VIRTUAL CABLE

Installation Manual, UDS Enterprise 4.0

INDEX

1. INTRODUCTION 3
1.1 Enterprise & Evaluation Versions 4
1.2 Features 5
1.3 Platform Architecture with UDS Enterprise 6
1.3.1 Network Architecture 8
1.4 Components of UDS Enterprise 11
1.4.1 UDS Server 12
1.4.2 UDS Tunnel 13
1.4.3 UDS Dbserver 14
1.4.4 UDS Actor 15
1.4.5 UDS Client 16
2. BEFORE INSTALLING UDS 17
2.1 Installing on VMware vSphere 17
2.1.1 Virtual Platform Requirements 17
2.1.2 Network connections 18
2.2 Installing on oVirt 19
2.2.1 Virtual Platform Requirements 19
2.2.2 Network connections 20
2.3 Installation on Microsoft Hyper-V 21
2.3.1 Virtual Platform Requirements 21
2.3.2 Network connections 22
2.4 Installation on XenServer/XCP-ng 23
2.4.1 Virtual Platform Requirements 23
2.4.2 Network connections 24
2.5 Installation on Nutanix Acropolis 25
2.5.1 Virtual Platform Requirements 25
2.5.2 Network connections 26
2.6 Installation on OpenStack 27
2.6.1 Virtual Platform Requirements 27
2.6.2 Network connections 28
2.7 Installation on OpenNebula 29

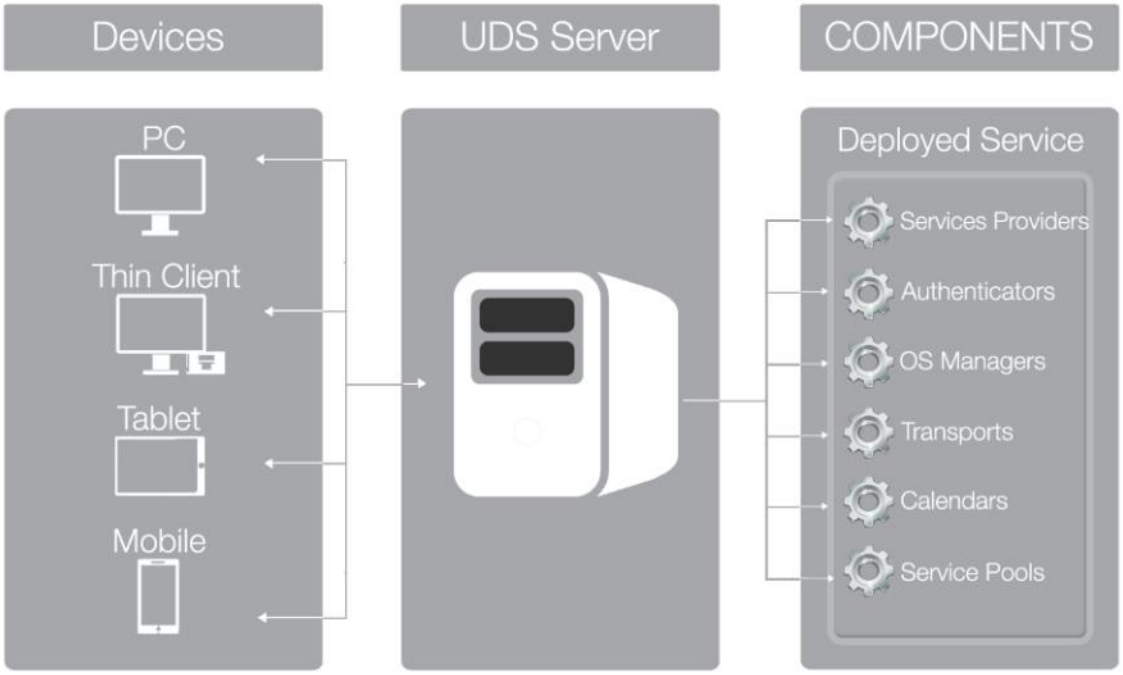
2.7.1 Virtual Platform Requirements 29
2.7.2 Network connections 30
3. INSTALLING UDS ENTERPRISE 31
3.1 UDS Enterprise Environment Requirements 31
3.1.1 Infrastructure requirements 31
3.1.2 Network Requirements 31
3.2 UDS Enterprise Component Installation 32
3.2.1 UDS Dbserver 32
3.2.2 UDS Server 36
3.2.3 UDS Tunnel 56
3.2.4 UDS Actor 74
3.2.5 UDS Client 102
4. ABOUT VIRTUAL CABLE 114

1. INTRODUCTION

UDS Enterprise is a multiplatform connection broker that manages user access to different types of services, including dedicated virtual desktops, desktop sessions / virtual application sessions, remote access to physical / virtual computers, etc...

UDS Enterprise offers a set of software elements that make up a platform for the management of the life cycle, administration and deployment of desktop services.

This document contains the basic instructions for installing UDS Enterprise software elements on an existing virtual infrastructure.



1.1 Enterprise & Evaluation Versions

There are different versions of UDS software, each of them oriented to different uses and scenarios:

- **UDS Enterprise:** Ideal for VDI environments of any size, it allows technical support in case of any doubt or incident with the software and updates to new versions. Subscription purchase required.

- **UDS Evaluation:** Pensada para la realización de Pilotos, PoCs y en general pruebas con una duración limitada (60 días). Por defecto 5 usuarios ampliables bajo petición.
  No requiere adquisición de suscripción

UDS software uses a database to store user statistical data and configuration parameters regarding the system. For this function, UDS supports MySQL from its version **8.0.1** and MariaDB **10.6.1** (other MySQL-based databases would also be supported).

In the UDS Enterprise version, the database must be external. If it does not have one, Virtual Cable may supply one in virtual appliance format, not being included in the UDS software support.

UDS Enterprise Evaluation also supports an external Database, although to facilitate its deployment it is possible to enable an internal one;

NOTE:
If an internal database is used, it will not be possible to migrate to new versions.

The main differences between the available UDS versions are shown in the following comparison table:

	Enterprise	Evaluation
Number of users	Up to unlimited	5
Duration	Up to unlimited	60 days
¿Database?	Requires external	External/Internal
¿WAN tunneling of connections?	Yes	Yes
¿Support?	Yes	No
¿Premium Support?	Yes	No

1.2 Features

Among the main features of UDS Enterprise it is worth highlighting:

- Deployment, Easy installation and management
- Automated virtual desktop deployment and remote writer's session access management
- Virtualization of Windows application sessions for users of Windows/Linux environments using Remote Desktop Services (RDS)
- Virtualization of Linux application sessions for users of Windows/Linux environments using X2Go
- Multi-hipervisor, currently supports VMware vSphere, KVM OLVM/oVirt, Microsoft Hyper-V, Citrix XenServer/Citrix Hypervisor, VMware vCloud, Microsoft Azure, Amazon AWS, Google Cloud, OpenGnsys, OpenNebula, OpenStack, Proxmox, Nutanix AHV, Nutanix Prims Central, XCP-ng and Scale Computing.
- Multi-authenticator. Allows you to define users and user groups from different external sources and with virtually unlimited configurations
- Multi-connector authentication systemj: Active Directory, eDirectory, Azure AD, RADIUS, OpenLDAP, SAML2, OAuth2, CAS, Internal Authentication System, Device Authentication System, IP …
- Generation of reports on the status and use of the platform
- Task scheduling system (deployment of services, user access control, etc...) through calendars
- Secure WAN access for virtual desktops and applications using an SSL tunneler included in the suscription
- Full customization of the login portal and user service pages
- Product roadmap based on customer and community requests
- Subscription cost model that entitles UDS Enterprise support, new versions, updates and patches
- Non-redistributable subscription model by user brackets segmented into verticals.

1.3 Platform Architecture with UDS Enterprise

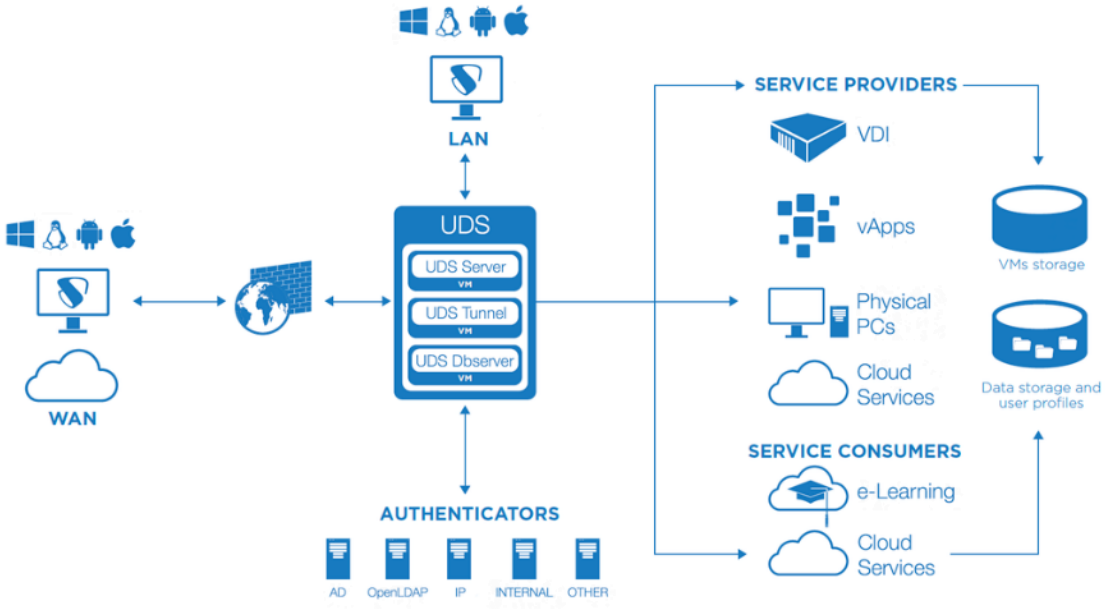
An optimal design of a desktop services platform is a fundamental part of achieving all the benefits that architecture can provide. Each layer that makes up this architecture must be designed in such a way that it fulfills its function without penalizing the rest.

The main elements that make up an architecture with UDS Enterprise are:

- **Connection Clients:** They are access devices to virtual desktops and applications, such as: thin clients, zero clients, PCs, etc... It is important to identify whether desktop access will be from a LAN or from a WAN
- **UDS Servers:** They consist of a database to store all the data related to the environment, a connection broker that will manage the life cycle of desktop services and communication with hypervisors and other service providers; and a tunnel server to allow secure access from the outside. All of these items are served in virtual appliance format
- **Authenticators:** Active Directory servers, OpenLDAP, eDirectory, etc... which through its integration with UDS Enterprise will control user access to desktop services. Depending on the environment, we will have from one to an unlimited number of authenticators
- **Service Providers**
    - **Hypervisor platform:** In charge of executing the tasks of creation, power-on and deletion of virtual desktops managed from the broker. UDS Enterprise integrates with hypervisors such as: Microsoft Hyper-V, VMware vSphere, KVM (oVirt, OLVM, Proxmox, OpenStack and OpenNebula), Citrix XenServer and Nutanix Acropolis, etc...
    - **RDS Applications:** In charge of providing the Windows desktop or application sessions that will be managed by UDS Enterprise
- **Storage:** They will host the servers, virtual desktops, remote desktops, applications and/or other services on the platform. The choice of storage type is an important part of the design. Depending on the needs demanded by users in desktop services, we must select the most appropriate type in terms of performance

With a clear idea of the architecture design, it will be time to start scaling the platform, taking into account the number of users who will access it.

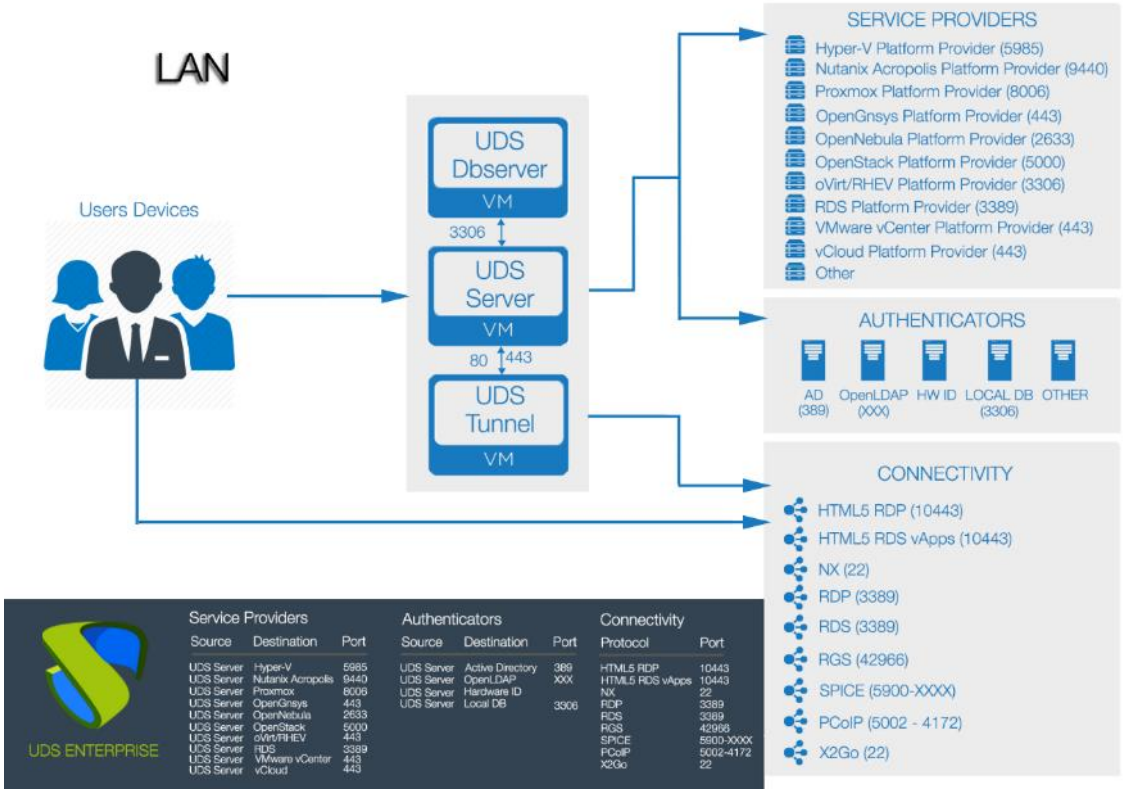
In the following image you can see an example of a VDI architecture with UDS Enterprise:



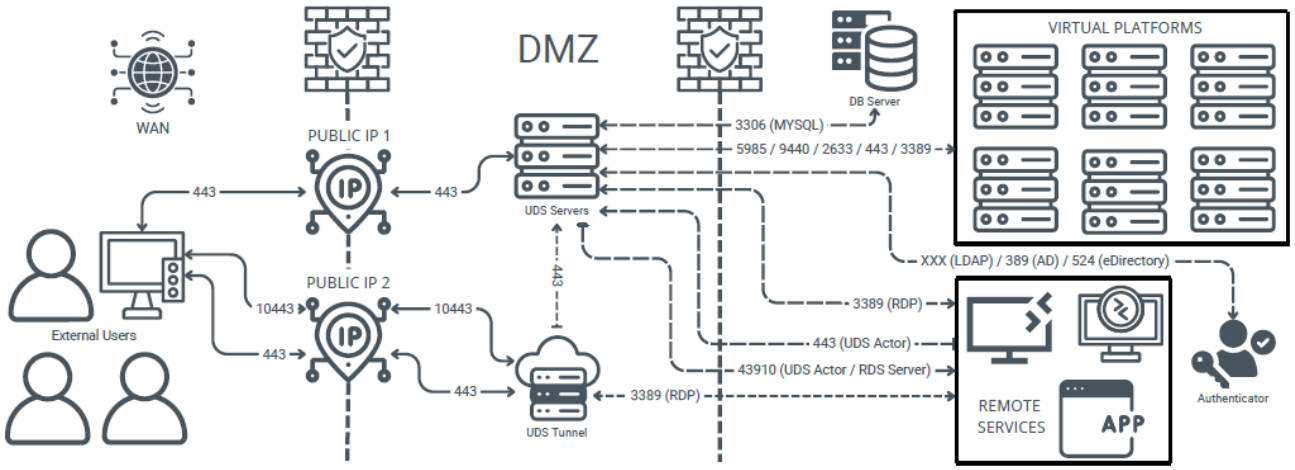
1.3.1 Network Architecture

UDS Enterprise can be configured to be accessible by users located on a local network or users coming from a WAN (internet) without the need for VPN or LAN_extension.

Example of deploying desktop services and virtual applications for user access over a LAN (deployable to users accessing from a VPN or LAN-extension):



Example of deploying desktop services and virtual applications for user access over a WAN (internet).



In order to publish UDS on the internet and for its services to be accessible by users, two public IP addresses will be needed, one for UDS Server and one for UDS Tunnel (it is possible to perform this process with a single public IP address by changing the default ports and configuring internal NAT).

Safe Employment Procedure:

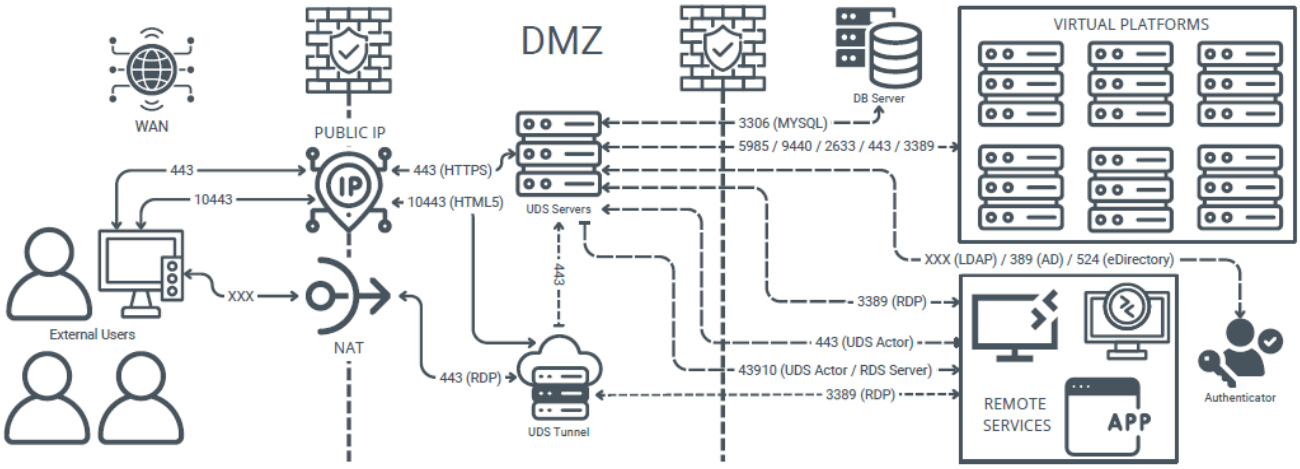
It is recommended to open only the ports strictly necessary for the correct functioning of UDS Enterprise in communication between its components.

It is recommended to install UDS Server and UDS Tunnel in the DMZ and UDS DBServer in the server zone. All these zones must be delimited by firewalls.

In the case of accessing the UDS Server from the WAN, the use of port 443 is recommended to force the use of HTTPS in the UDS web portal.

The use of HTTPS requires the use of valid web certificates, it is the customer's responsibility to provide and install such certificates.

Example with 1 single public IP (used for UDS server and Tunnel):



Ports required for the different elements and services of a VDI environment:

UDS Appliances Internal Use

SOURCE	DESTINATION	PORT
UDS Server	DB Server	3306
UDS Tunnel	UDS Server	443
UDS Server	UDS Actor	43910
UDS Actor	UDS Server	443

Authenticator

SOURCE	DESTINATION	PORT
UDS Server	Active Directory	389 / 636 (SSL)
UDS Server	LDAP	- - - -
UDS Server	Internal DB	3306
UDS Server	RADIUS	2633
UDS Server	SAML	443
UDS Server	Azure AD	443
UDS Server	Hardware ID	- - - -

VDI Connectivity

SOURCE	DESTINATION	PORT
UDS Server	HTML5 RDP	3389
UDS Server	No Machine NX	22
UDS Server	RDP	3389
UDS Server	RDS	3389
UDS Server	SPICE	2633
UDS Server	PCOIP	443
UDS Server	X2GO	22
UDS Server	HTML5 SSH	22
UDS Server	Nice DCV	8443

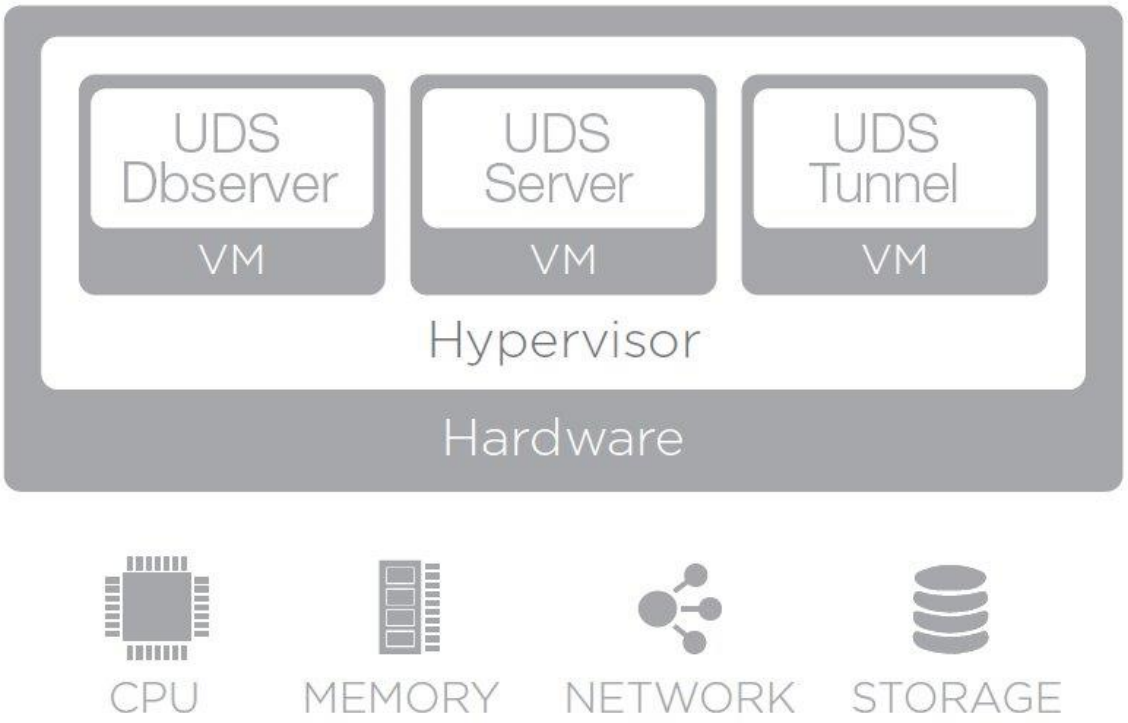
Virtual Platforms

SOURCE	DESTINATION	PORT
UDS Server	Proxmox	8006
UDS Server	Hyper-V	5985
UDS Server	Nutanix AHV	94440
UDS Server	OpenNebula	2633
UDS Server	OpenStack	2633
UDS Server	oVirt/OLVM	443
UDS Server	RDS	3389
UDS Server	VMware vCenter	443
UDS Server	Microsoft Azure	443
UDS Server	AWS	443
UDS Server	vCloud	443
UDS Server	Huawei	443
UDS Server	OpenGnsys	443
UDS Server	XenServer	443

1.4 Components of UDS Enterprise

UDS Enterprise is made up of 5 elements that interact with each other:

- **UDS Server:** Installs as a virtual machine (VM) and is delivered as a virtual appliance.
- **UDS Tunnel:** Installs as a VM and is provided as a virtual appliance.
- **UDS Dbserver:** It is installed as a VM and is provided in virtual appliance format (Optional for UDS Evaluation).
- **UDS Actor:** It is installed as a service on the VM to be used as a template for deploying desktop pools, on RDS application servers to provide virtualized applications, and on static machines to control user session usage.
- **UDS Client:** Se instala en el equipo cliente para poder conectar con los servicios de escritorio (Con el tipo de conexión HTML5 este componente no es necesario).



The characteristics and technical requirements of each of them are defined below:

1.4.1 UDS Server

It is the software that mediates between connection clients and service providers. It is the fundamental piece of UDS, it performs the functions of a connection broker to desktop services, allowing the administration and management of the platforms defined as implemented services.

Virtual Appliance with the following features:

- Virtual Disk: 16 GB
- Memory: 4 GB
- CPU: 2 vCPU
- Network: 1 vNIC

Requirements:

- 1 IP Direction
- Name of Server
- Netmask
- IP Gateway
- IP DNS
- Domain Name (optional)
- IP or database server name
- Database instance name and port
- User and password with database instance permissions
- Activation Code (Enterprise or Evaluation)

Secure use procedure: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.

1.4.2 UDS Tunnel

Software responsible for making secure connections from a connection client to desktop services over the WAN. It can also provide this access via HTML5 (based on RDP).

The UDS tunneler allows you to connect from any device/browser/client with Windows, Linux and MacOS OS to desktop services and applications through an SSL tunnel with external-to-end encryption.

Virtual Appliance with the following features:

- Storage: 20 GB
- Memory: 4 GB
- CPU: 2 vCPU
- Network: 1 vNIC

Requirements:

- 1 IP Direction
- Server Name
- Máscara de red
- IP Gateway
- IP DNS
- Domain Name (optional)
- UDS Server Name with Recognized Valid Certificate
- User with UDS Server administrator permissions

1.4.3 UDS Dbserver

It is the component that is responsible for storing all the data in the UDS system: service providers, authenticators, connectivity, etc. and all the information that will make it possible to generate statistics.

MySQL database manager is supported from version **8.0.1** and MariaDB **10.6.1**.

You need to have a database at installation time properly configured with a valid instance and a user with permissions. This will be the first component to be configured or have available.

¡IMPORTANT!

If you do not have such a database manager, Virtual Cable can provide this component as a virtual appliance. This component is not included in UDS Enterprise support.

Virtual Appliance with the following features:

- Storage: 24 GB
- Memory: 4 GB
- CPU: 2 vCPU
- Network: 1 vNIC

Requirements:

- 1 IP Direction
- IP DNS
- Server Name
- Netmask
- IP Gateway
- Domain Name (opcional)
- DB instance name
- User with permissions on the instance

1.4.4 UDS Actor

It is the software that performs the communication and interface functions for the transmission of data (virtual desktop status, machine name...) and commands between UDS Server and the desktop services managed by UDS.

It is installed as a service on the virtual machine to be used as a template (gold image) to generate groups of desktop services, on Remote Desktop Services (RDS) servers to provide desktop sessions and virtualized applications, and on static machines to handle user sessions.

The operating systems supported to generate virtual desktops are:

- Windows 11
- Windows 10
- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Linux (Debian, Ubuntu, Fedora, OpenSuse, etc…)

The operating systems supported to generate Windows virtual applications are:

- Windows Server 2022
- Windows Server 2019
- Windows Server 2016

The operating systems supported to generate Linux virtual applications are:

- Ubuntu / Debian

Requirements:

- .Net Framework 3.5 SP1 (Windows Machines)
- Python 3.9 (Linux Machines)
- IP or UDS Server Name
- Username and password with UDS Server administration permissions

1.4.5 UDS Client

It is the software that makes the call to the connection protocol to connect to virtual desktops and applications.

It is installed on the client computer from which the connection to the desktop services is to be made.

The supported operating systems are:

- Windows 11
- Windows 10
- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Linux (Debian, Ubuntu, Fedora, OpenSuse, etc…)
- MAC OS (Versions 13 and 14)

2. BEFORE INSTALLING UDS

UDS Enterprise components can be hosted on different virtualization platforms.

Although the UDS Enterprise components are hosted on a single virtual platform, UDS is capable of managing the deployment of virtual desktops on multiple virtual platforms, which are completely independent of the virtual platform where UDS is hosted.

The contents of this section describe the requirements for installing UDS Enterprise on different virtualization platforms and the requirements of the virtual platform on which the software is to be installed.

2.1 Installing on VMware vSphere

2.1.1 Virtual Platform Requirements

UDS Enterprise can be deployed on VMware vSphere 7 or higher platforms.

To find out the requirements of a VMware vSphere platform, you can consult the manufacturer's documentation.

The VMware platform on which UDS will be deployed must meet the following requirements:

- At least one VMware ESXi server with a valid license is required to host the UDS servers and generate the virtual desktops
- The vSphere platform must be managed by a vCenter with a valid license
- In order for UDS Enterprise to be able to integrate and send requests to vCenter for them to be executed, it is necessary to have the credentials of a user with the necessary permissions on the VMware vSphere platform on which the virtual desktops and applications are going to be deployed
- It is necessary to have defined at least one Virtual Machine Port Group to which the virtual servers of the UDS platform are going to be connected
- It is necessary to have defined at least one Virtual Machine Port Group to which the different virtual desktops managed by UDS are going to be connected
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS

At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS

2.1.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	vCenter	443
UDS Server	Autenticathor	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

2.2 Installing on oVirt

2.2.1 Virtual Platform Requirements

UDS Enterprise can be deployed on oVirt 4.x platforms

The oVirt platform on which UDS will be deployed must meet the following requirements:

- At least one oVirt node server is needed to host the UDS servers and generate the virtual desktops
- The oVirt platform needs to be managed by an oVirt-engine
- In order for UDS to be installed and to be able to send requests to oVirt-engine and for them to be executed, it is necessary to have the credentials of a user with administration permissions on the oVirt platform on which the virtual desktops are going to be deployed
- It is necessary to have at least one cluster defined to create and configure the different virtual desktops managed by UDS
- It is necessary to have defined at least one "Logical network" to which the virtual servers of the UDS platform are going to be connected
- It is necessary to have at least one "Logical Network" defined to which the different virtual desktops managed by UDS are going to be connected
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS
- At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS

2.2.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	oVirt-Engine	443
UDS Server	Authenticator	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	80/443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

2.3 Installation on Microsoft Hyper-V

2.3.1 Virtual Platform Requirements

UDS can be deployed on Microsoft Hyper-V version 3 platforms.

The Microsoft Hyper-V platform on which UDS will be deployed must meet the following requirements:

- At least one Microsoft Hyper-V server with a valid license is required to host the UDS servers and generate the virtual desktops
- It is necessary to have defined at least one Virtual Switch to which the virtual servers of the UDS platform are going to be connected
- It is necessary to have defined at least one Virtual Switch to which the different virtual desktops managed by UDS are going to be connected
- It is necessary to have the credentials of a user with administration permissions on the Microsoft Hyper-V platform on which the virtual desktops are to be deployed
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS
- At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS
- The machines that are used as template machines (Gold Image) must be of the type: Generation 1
- For the correct operation of Microsoft Hyper-V with UDS it is necessary to run the following script (as administrator) on all Hyper-V hosts that are going to be used by UDS (even if they are part of a Hyper-V cluster):

```
Invoke-Expression((New-Object
System.Net.Webclient).DownloadString('https://images.udsenterprise.com/files/hyperv/EnableRemoting.ps1'))
```

2.3.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	Hyper-V	443
UDS Server	Authenticator	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

2.4 Installation on XenServer/XCP-ng

2.4.1 Virtual Platform Requirements

UDS Enterprise can be deployed on XenServer/XCP-ng platforms from version 8.

The XenServer/XCP-ng platform on which UDS will be deployed must meet the following requirements:

- At least one XenServer/XCP-ng server is needed to host the UDS servers and generate the virtual desktops
- In order for UDS to be installed and to be able to send requests to XenServer/XCP-ng and for them to be executed, it is necessary to have the credentials of a user with administration permissions on the XenServer/XCP-ng platform on which the virtual desktops are to be deployed
- It is necessary to have defined at least one network to which the virtual servers of the UDS platform are going to connect
- It is necessary to have at least one network defined to which the different virtual desktops managed by UDS are going to be connected
- The storage used to host the virtual desktops must be of type SR
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS
- At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS

2.4.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	XenServer/XCP-ng	443
UDS Server	Authenticator	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

2.5 Installation on Nutanix Acropolis

2.5.1 Virtual Platform Requirements

UDS Enterprise can be deployed on Nutanix AHV platforms.

The Nutanix AHV platform on which UDS will be deployed must meet the following requirements:

- In order for UDS to be installed and to be able to send requests to AHV and for them to be executed, it is necessary to have the credentials of a user with administration permissions on the AHV platform on which the virtual desktops are going to be deployed
- It is necessary to have defined at least one network to which the virtual servers of the UDS platform are going to connect
- It is necessary to have at least one network defined to which the different virtual desktops managed by UDS are going to be connected
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS
- At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS

2.5.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	AHV	443
UDS Server	Authenticator	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

2.6 Installation on OpenStack

2.6.1 Virtual Platform Requirements

UDS Enterprise can be deployed on OpenStack platforms starting with the Stein version.

The OpenStack platform on which UDS will be deployed must meet the following requirements:

- In order for UDS to be installed and to be able to send requests to OpenStack and for them to be executed, it is necessary to have the credentials of a user with administration permissions on the platform
- It is necessary to have defined at least one network to which the virtual servers of the UDS platform are going to connect
- It is necessary to have at least one network defined to which the different virtual desktops managed by UDS are going to be connected
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS
- At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS

2.6.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	OpenStack	5000
UDS Server	Authenticator	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

2.7 Installation on OpenNebula

2.7.1 Virtual Platform Requirements

UDS Enterprise can be deployed on OpenNebula 5.x platforms

The OpenNebula platform on which UDS will be deployed must meet the following requirements:

- In order for UDS to be installed and to be able to send requests to OpenNebula and for them to be executed, it is necessary to have the credentials of a user with administration permissions on the platform
- It is necessary to have at least one network to which the virtual servers of the UDS platform are going to connect
- It is necessary to have at least one network defined to which the different virtual desktops managed by UDS are going to be connected
- At least 60 GB of free disk space is necessary to host the virtual servers that make up UDS
- At least 12 GB of free RAM are necessary to host the virtual servers that make up UDS

2.7.2 Network connections

It is necessary to have the following communication ports enabled between the different elements that make up the UDS platform:

Origin	Destiny	Port
UDS Server	UDS MySQL	3306
UDS Server	OpenNebula	2633
UDS Server	Authenticator	389, 636, XXX
UDS Server	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Server	UDS Tunnel	443
UDS Server (Actor)	Virtual desktops	43910
Virtual desktops	UDS Server (Actor)	443
UDS Tunnel	Virtual desktops	3389 (RDP), 22 (NX), XXX
UDS Tunnel	UDS Server	443
Users	UDS Server	80/443
Users (LAN)	Virtual desktops	3389 (RDP), 22 (NX), XXX
Users	UDS Tunnel	443
Users	UDS Tunnel (HTML5)	10443

3. INSTALLING UDS ENTERPRISE

At this point we will detail the installation of UDS Enterprise components and their requirements. The installation procedure will be the same for the different virtualization platforms (VMware vSphere, Microsoft Hyper-V, XenServer, etc...) supported by UDS.

3.1 UDS Enterprise Environment Requirements

3.1.1 Infrastructure requirements

The infrastructure requirements necessary for UDS to be deployed are:

- **Virtualization Platform.** It will be in charge of hosting the UDS servers, the virtual desktops generated and the application servers.
    - Virtualization platform manager username and password with permissions

Secure use procedure: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.

- **DNS Server.** This service is necessary both for the proper functioning of the virtual platform and the UDS environment to be deployed.
- **DHCP Server.** A DHCP server is required to assign IP addresses to the virtual desktop pools created by UDS.

3.1.2 Network Requirements

For the UDS network configuration it is necessary to have at least 3 IP addresses (Server, Tunnel and database).

It is also necessary to have available:

- Netmask
- DNS server IP address
- Gateway IP Address
- Domain Name (if any)
- IP address or name of the virtualization platform manager

NOTE: If you install UDS Evaluation, you can do without the "database" component. In this case, only 2 IP addresses (Server and Tunnel) will be required.

3.2 UDS Enterprise Component Installation

3.2.1 UDS Dbserver

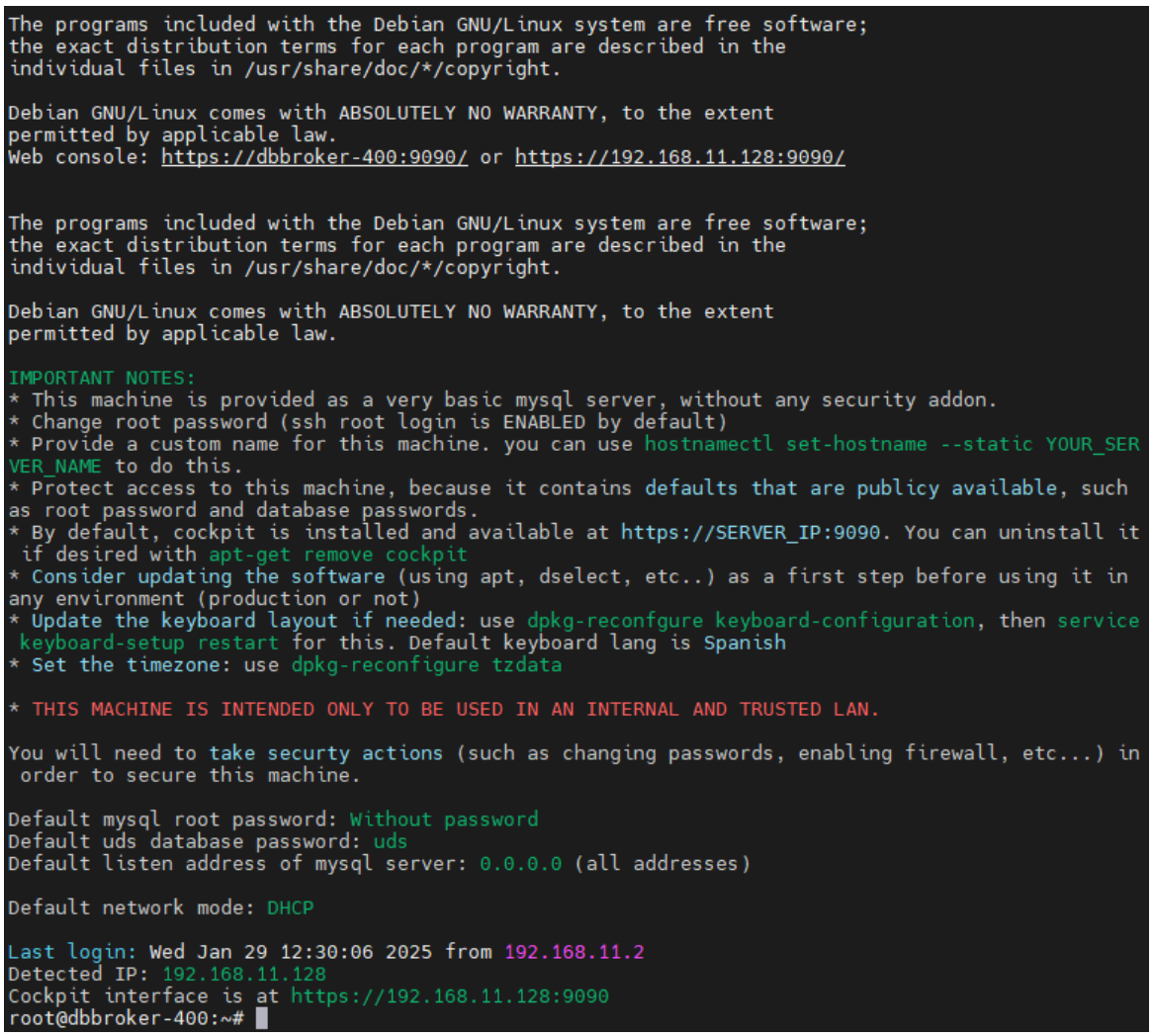
Remember that in case of installing UDS Evaluation, the database can be configured internally in the UDS Server component (although you will lose the system update option).

If you are using the database virtual appliance provided by Virtual Cable, you would have to perform the following tasks:

Access the database server with the following credentials:

- **User**: root
- **Password**: uds

NOTE: It is recommended to modify the default password to provide the system with greater security. You can use the command: *passwd. (passwords* They must be of sufficient length and include uppercase, lowercase, numbers, and special characters*...)*



Configure the network parameters of the virtual machine. To do this, the "***interfaces***" file will be modified, and a static IP address is assigned to it (by default the virtual appliance is configured by **dhcp**).

```
root@dbserver:~# nano /etc/network/interfaces
```

Depending on the virtualization platform we use to host the "database" component, we need to assign the new static IP address to the corresponding network interface (usually it will always be "eth0"):

```
  GNU nano 5.4                                /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 192.168.11.70
        netmask 255.255.255.0
        gateway 192.168.11.1

allow-hotplug enp1s0
iface enp1s0 inet dhcp

allow-hotplug ens32
iface ens32 inet dhcp
```

It is also necessary to review and modify, if necessary, the "***resolv.conf***" file to configure the DNS servers:

```
root@dbserver:~# nano /etc/resolv.conf
```

```
  GNU nano 5.4                                /etc/resolv.conf *
nameserver 192.168.11.1
nameserver 8.8.8.8
```

Once all the necessary network data has been configured, we restart the server and check that all the data has been configured correctly:

```
Debian GNU/Linux 12 dbbroker-400 tty1

Web console: https://dbbroker-400:9090/ or https://192.168.11.70:9090/

dbbroker-400 login:
```

Once the IP data of the server is configured, it would be available for use with UDS. By default, the database server has the following instance configured ready to use with the UDS server:

- **Instance**: uds
- **User**: uds
- **Password**: uds

NOTE: It is recommended to modify the password to provide the system with greater security. To perform this task we must execute within the MySQL console the command:

```
grant all on database_name.* to 'usuario'@'%' identified by 'new_password';
```

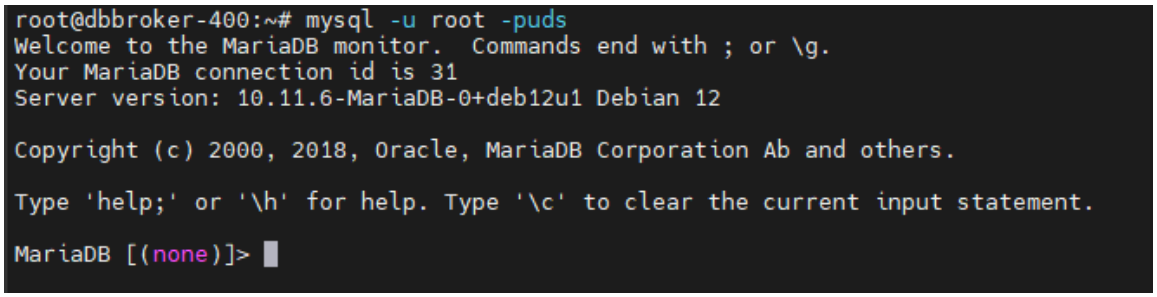
Secure use procedure: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.

Once these tasks are performed, the database will be available for use with the UDS Server component.

If it is necessary to create a new database instance for UDS, we would perform the following process:

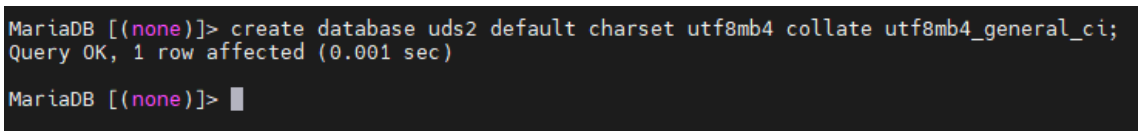
Access the MySQL service with the following credentials:

- **User**: root
- **Password**: uds



Create the new DB instance using the:

```
create database database_name default charset utf8mb4 collate utf8mb4_general_ci;
```



Where ***database_name*** will be the name of the new DB instance.

Create a user with administrator permissions on the new database instance using the command:

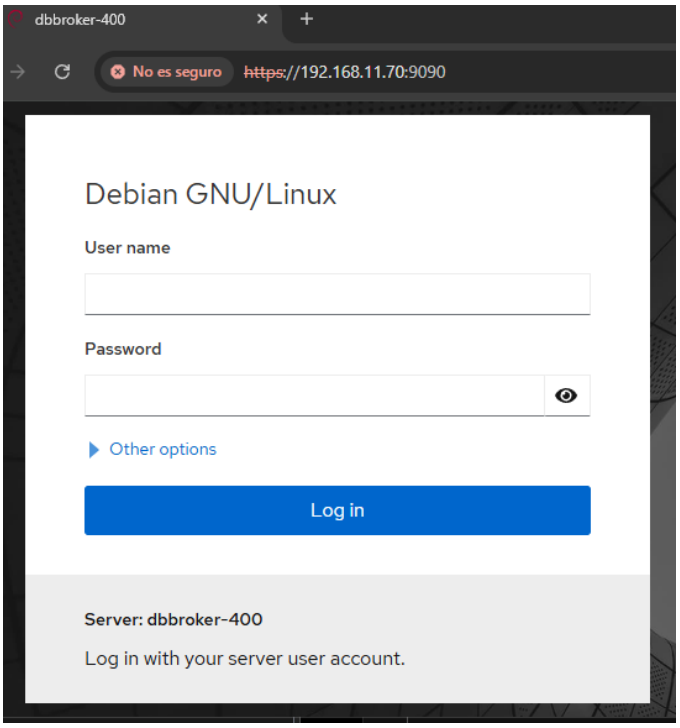
```
grant all on database_name.* to 'usuario'@'%' identified by 'password';
```

```
MariaDB [(none)]> grant all on uds2.* to 'uds2'@'%' identified by 'uds2';
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]>
```

Where ***database_name*** will be the name of the previously created DB instance, ***user*** will be the name of the new user with admin permissions on this DB instance and ***password*** the password assigned to the indicated user.

Secure use procedure: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.

By default, the database server includes the Cockpit tool, which allows you to manage and monitor certain components of the server graphically. To access, we will use the IP address or server name with port 9090:



Safe Employment Procedure:

In general, it is advisable to implement basic and essential security mechanisms on any server; strong passwords, backups, having security solutions, keeping systems updated and modifying the configurations, usernames and passwords included, by default.

In addition, for a database server, although in the case of UDS Enterprise, we do not store any confidential information, but configuration and registration information, it is important:

- Limit Access
- Encrypt information
- Monitor activity

It is advisable to disable SSH access to this server, so it is only accessible by console.

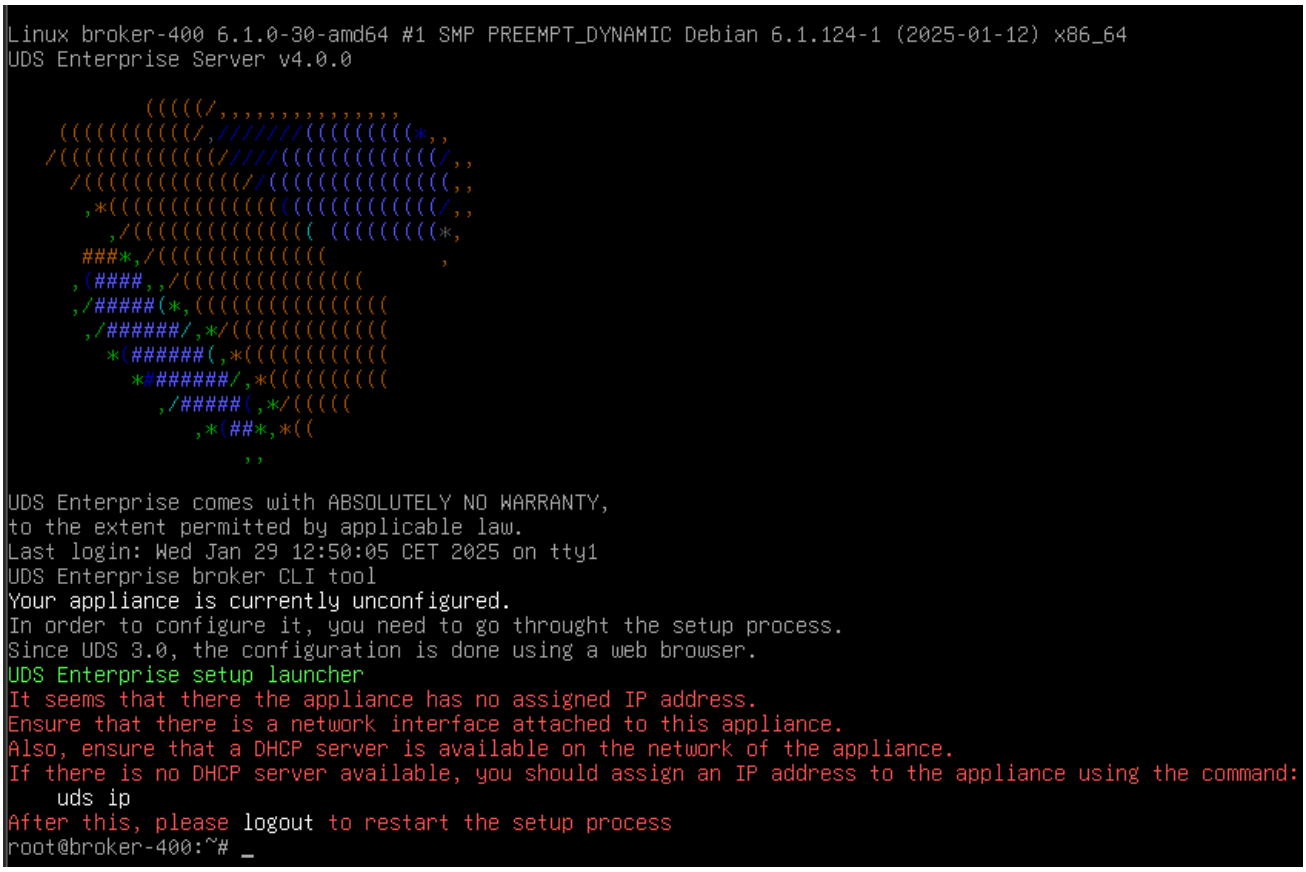
3.2.2 UDS Server

Once the UDS Server virtual appliance has been imported to a supported virtualization platform, we turn on the virtual machine to proceed with its initial configuration.

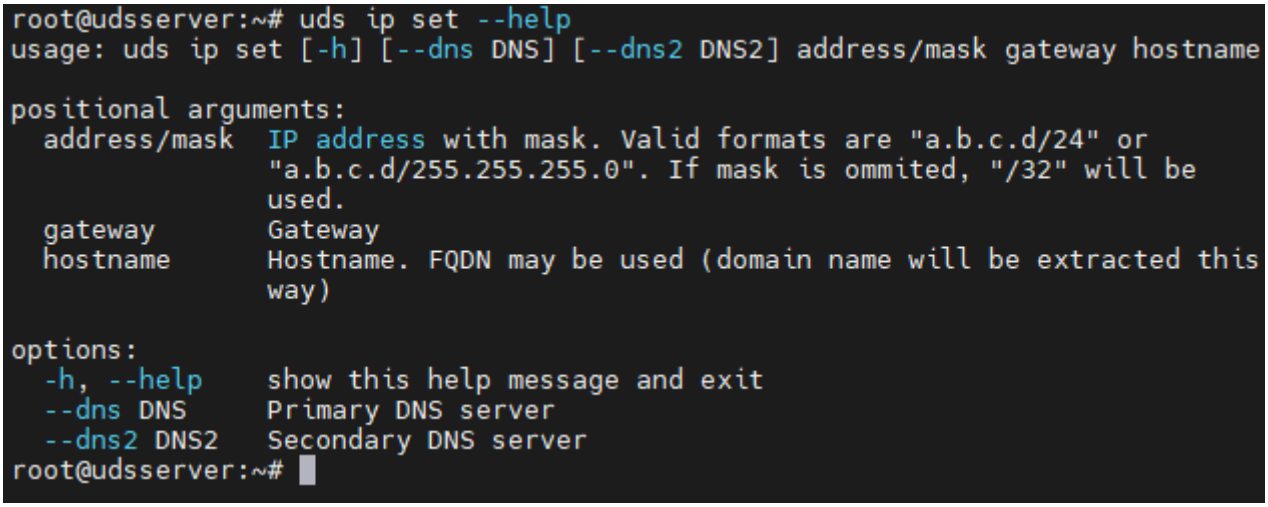
NOTE: In order to successfully configure a UDS server, it is necessary to have a database server configured with an available instance. If you use a database that has already been used with UDS and contains data from a previous UDS version, all data will be migrated for use with the new version (database migrations are only allowed from contiguous versions).

The UDS Evaluation version allows you not to use the external database, although you cannot perform updates or migrations between versions (in any version of UDS, Enterprise or Evaluation, it is always advisable to use an external database).

By default, the UDS Server virtual appliance will take a network configuration via DHCP. In case there is no server on the network that assigns IP addresses, we will have to assign the network data manually:



To do this, we use the ***uds ip set*** command with the configuration options:



Proceed with the manual configuration of the server's network data:

`uds ip set` *`ip_server/mask gateway name_server`*

Additionally, we can indicate the domain (extracted from the server name) and the DNS servers (with the --dns parameter)

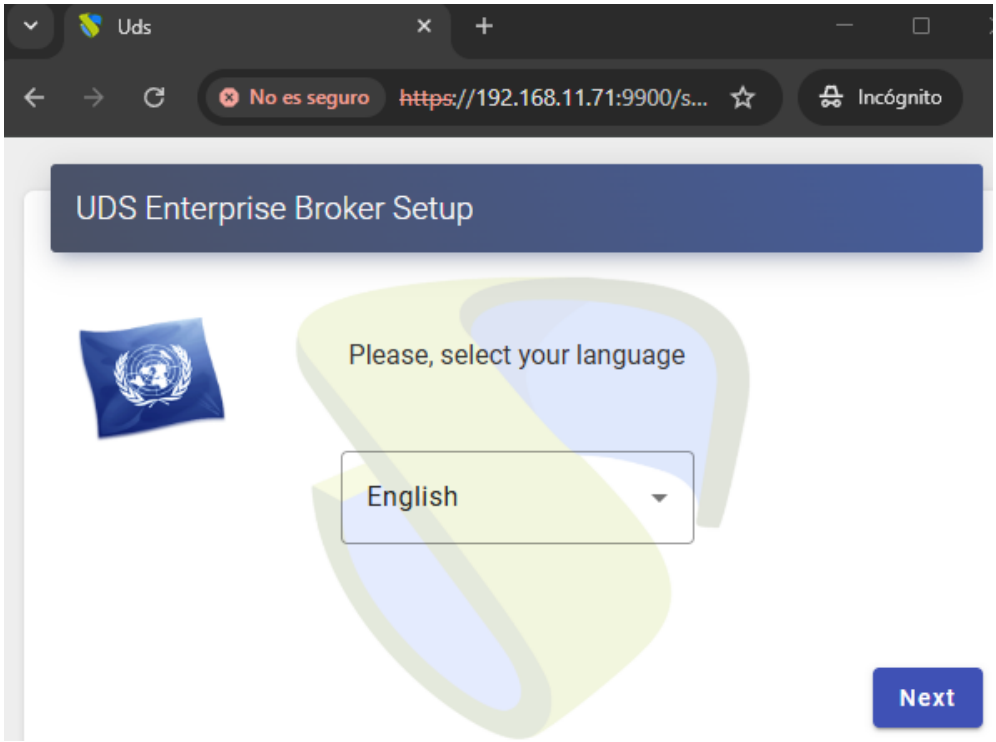
```
root@broker-400:~# uds ip set 192.168.11.71/24 192.168.11.1 udsserver.vc.local --dns 192.168.11.1
UDS Enterprise broker CLI tool
Updating network configuration...[  335.912526] e1000: eth0 NIC Link is Up 1000 Mbps Full Duplex,
[  335.912676] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
done
New network configuration
DHCP: no
Using interface: eth0
Hostname: udsserver
Domain: vc.local
Address: 192.168.11.71
Mask: 255.255.255.0
Gateway: 192.168.11.1
DNS: 192.168.11.1
Secondary DNS: 80.58.61.250
You need to reboot your appliance in order to fully activate the new configuration
root@broker-400:~#
```

Once the IP data is configured, we will restart the server to apply the changes.

If we already have an IP address assigned to the server, either by its manual configuration or by the automatic assignment of a DHCP server, we will proceed to the configuration of the UDS Server component.

To do this, access via web browser (**with https**) the IP address of the server with port 9900

```
UDS Enterprise setup launcher
Your appliance IP is 192.168.11.71. We are going to start the web setup process for you right now.
To configure your appliance, please go to this URL: https://192.168.11.71:9900
Note that, by default, UDS Appliance generates self signed certificates.
```



Safety note:

To carry out the initial basic configuration, UDS Server incorporates its own security mechanism.

To be able to perform the initial configuration we need two things:

- The command "uds setup" must be running on the server
- We need to know the value of our server's Setup Code.

The "uds setup" command is executed automatically the first time the server is turned on and can be executed manually at any time from the console.

The "uds setup" command stops automatically when the initial setup is finished or manually by running "uds setup -s".

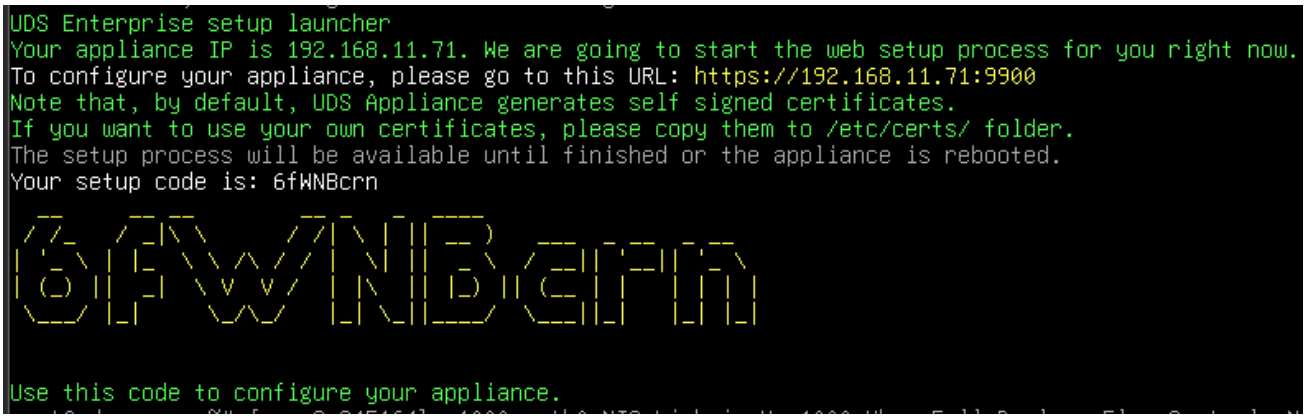
When you run the "uds setup" command, the "Setup Code" of your server appears on the screen, which will be requested if you want to perform the basic configuration.

The "Setup Code" is an eight-character token that **does NOT** circulate at any time on the network, which will be our "One-time pad".

"One-time pad" is part of an encryption technique that cannot be decrypted, but requires the use of a pre-shared, one-time key that is no smaller than the message being sent.

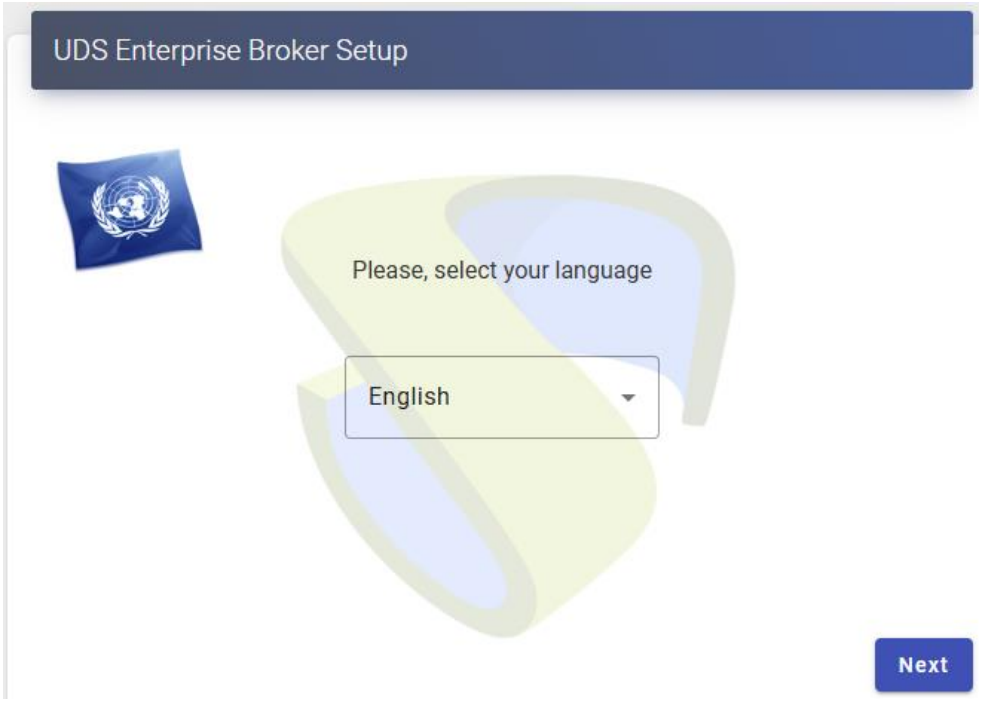
In this technique, plain text is combined with a random secret key, our one-time "One-time pad".

With this token at each end of the connection using a "salt", a set of random bits that we use as part of the cryptographic algorithm key, and using a 512 hash function, we obtain a 3,072-byte key, which is chopped up and used by both parties to encrypt and decrypt the information transmitted and received during this initial configuration. This key is never used in its entirety or reused in any of its parts.

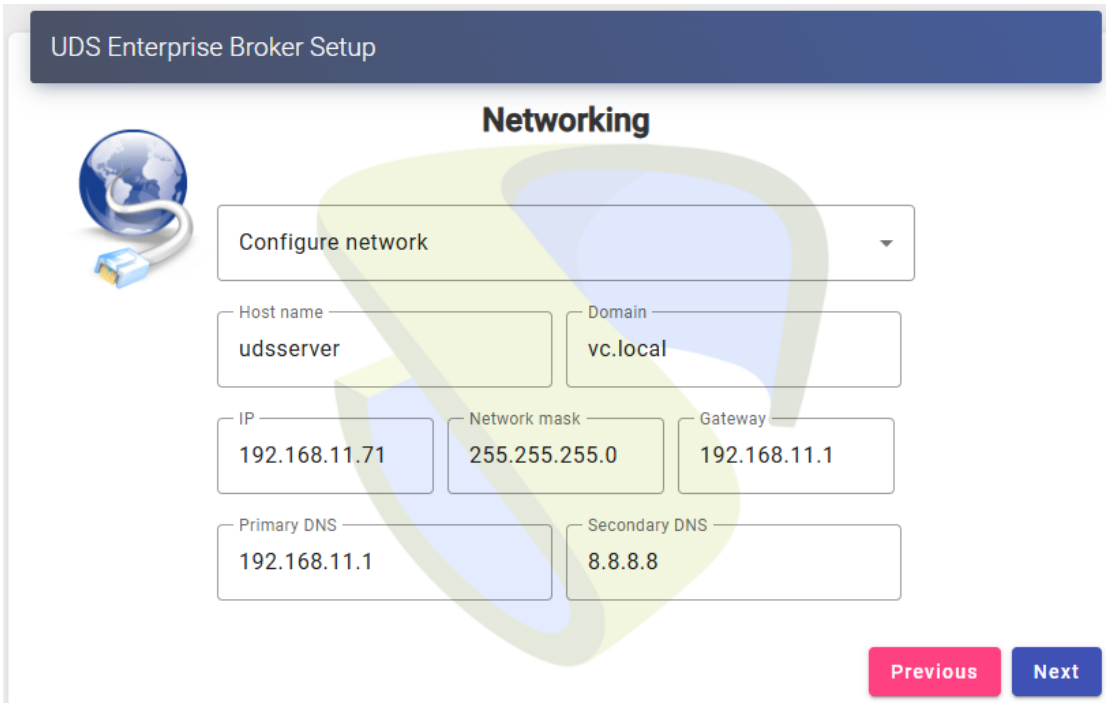


This "Setup Code" will be requested during the initial basic configuration.

Step 1.- Select the language of the configuration wizard:



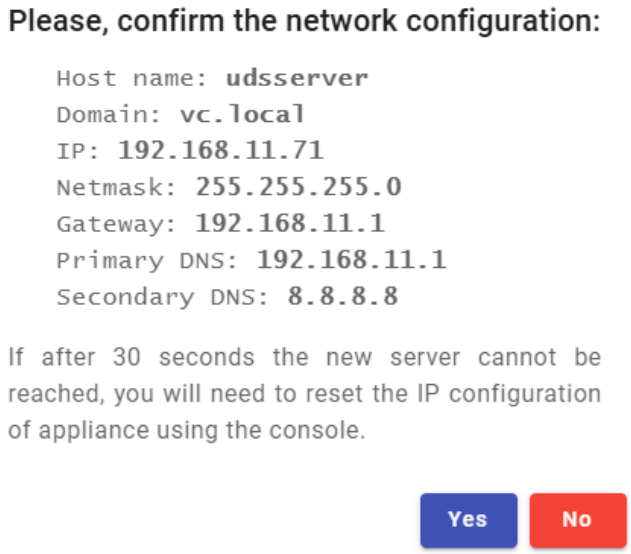
Step 2.- Select server name, domain (optional) and network data of the server:



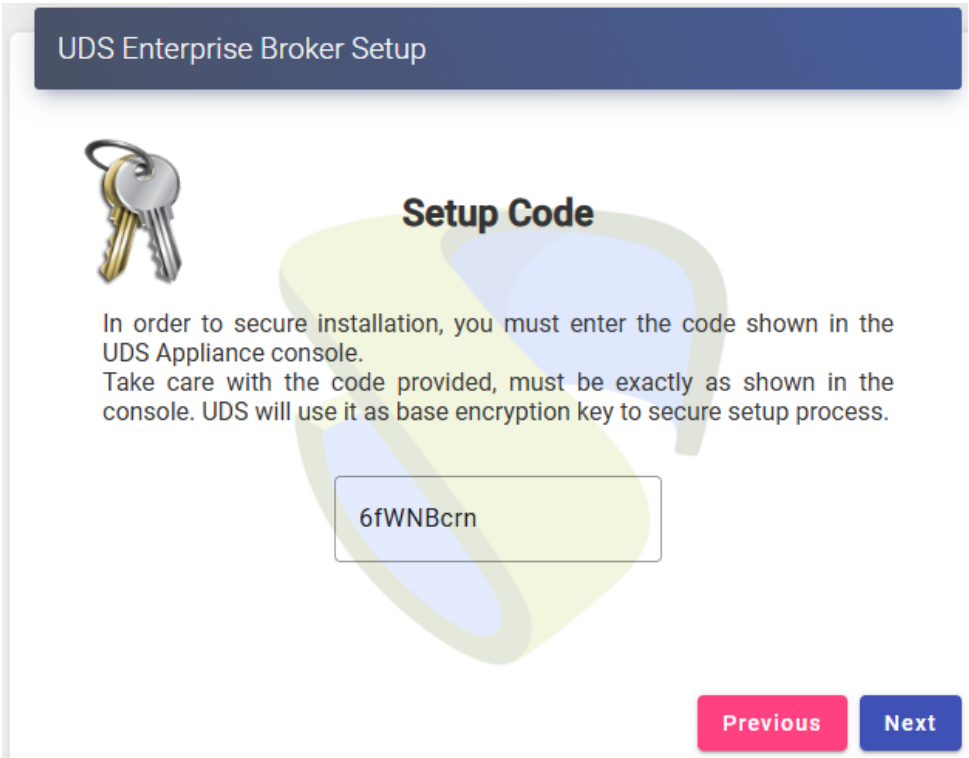
NOTE: If the IP address indicated is different from the one already configured by the server via DHCP or via manual configuration, the system will automatically redirect to the new address to continue with the configuration wizard.

NOTE: If all the data indicated is correct and you do not want to modify any data, you can use the option "**Skip network config (leave it as is)**".

Check that the data indicated are correct and we accept:

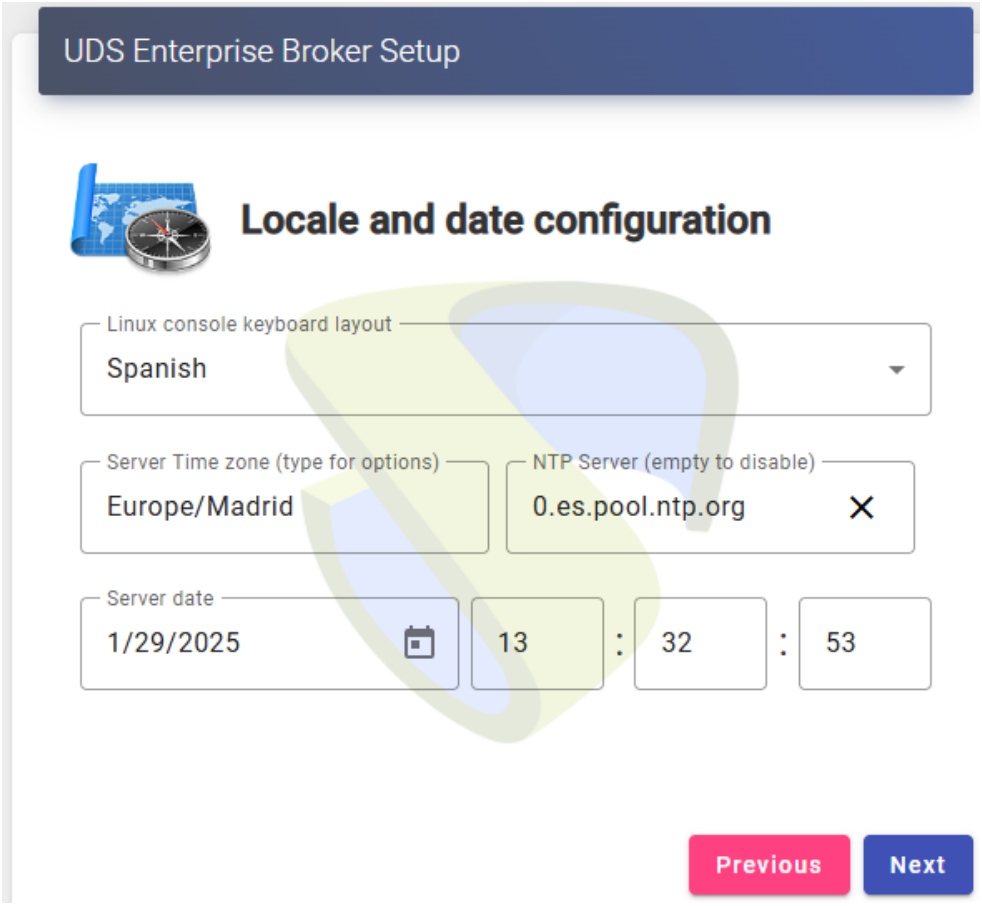


Step 3.- Add the security code ("Setup Code") that will appear in the console of our UDS Server appliance and that we saw at the beginning of this procedure:



If you do not have access to the server console, via ssh you can locate the code in the file: /etc/setupcode.uds

Step 4.- Configure the keyboard language that the server will have, the time zone and an NTP server (optional):



It is very important to select the time zone correctly, otherwise, there may be problems with certain functionalities (MFAs, authentication via SAML, etc...) and the system events will not show the time correctly.

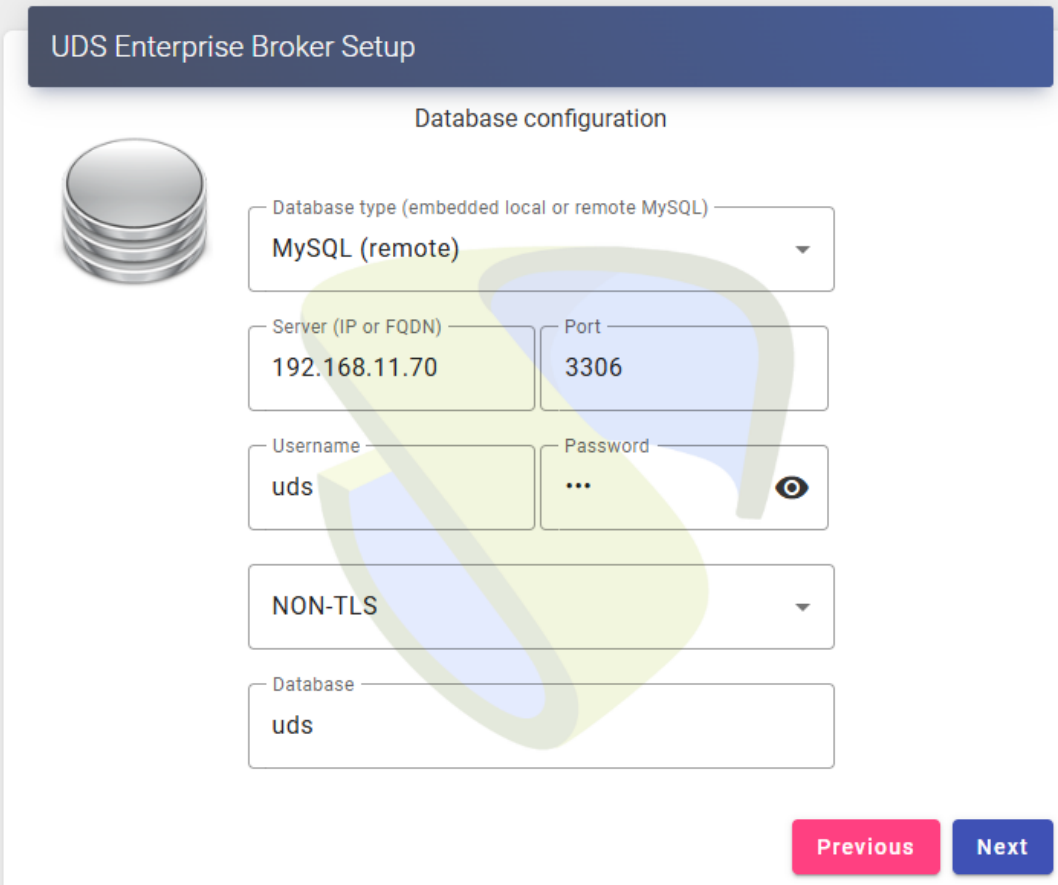
Step 5.- Select the type of database to use:

- **MySQL (remote)**: If we select this type, the system will require an external database server (valid and recommended for any edition of UDS).
- **Embedded (local):** If we select this type, the system will enable a local database on the UDS server. Not recommended for Enterprise versions or for Evaluation versions that need to be updated.

NOTE: If you select an on-premises database, you will not be able to upgrade or migrate the system with new versions without losing existing data.

If you select "***MySQL (remote)***", you must enter the connection data with the database:

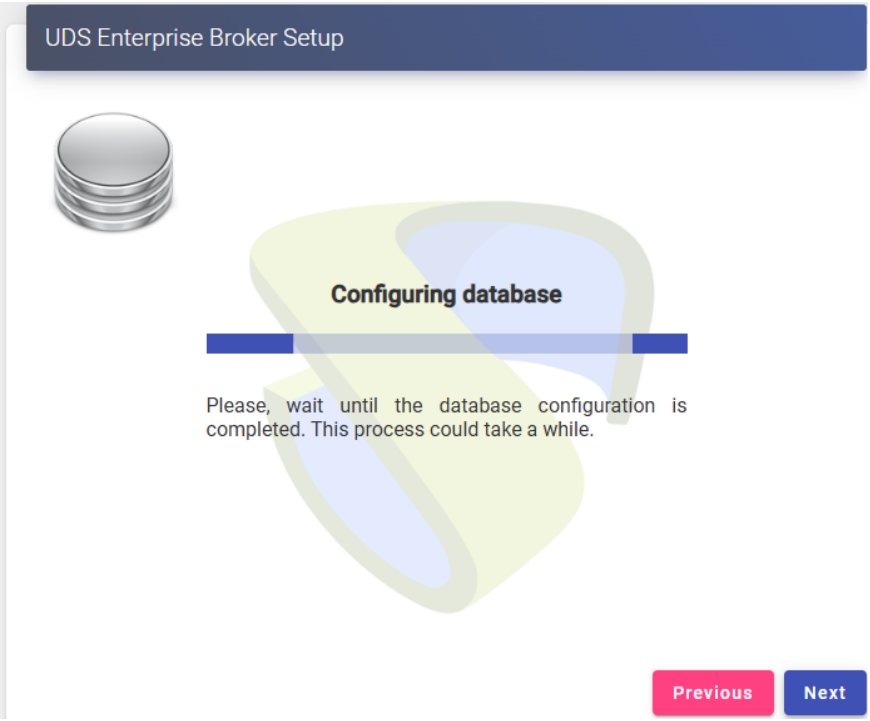
- **Server:** IP or database server name
- **Port:** Puerto de comunicación con el servidor de base de datos (por defecto: 3306)
- **Username:** Username with admin permissions on the DB instance.
- **Password:** User Password
- **Connection Type:** Type of connection to the database (if TLS is enabled, the database server has to support it, otherwise a non-TLS communication will be used)
- **Database:** DB instance name



NOTE: If you use the database server provided by UDS, the default data is: Username: uds // Password: uds // Database: uds

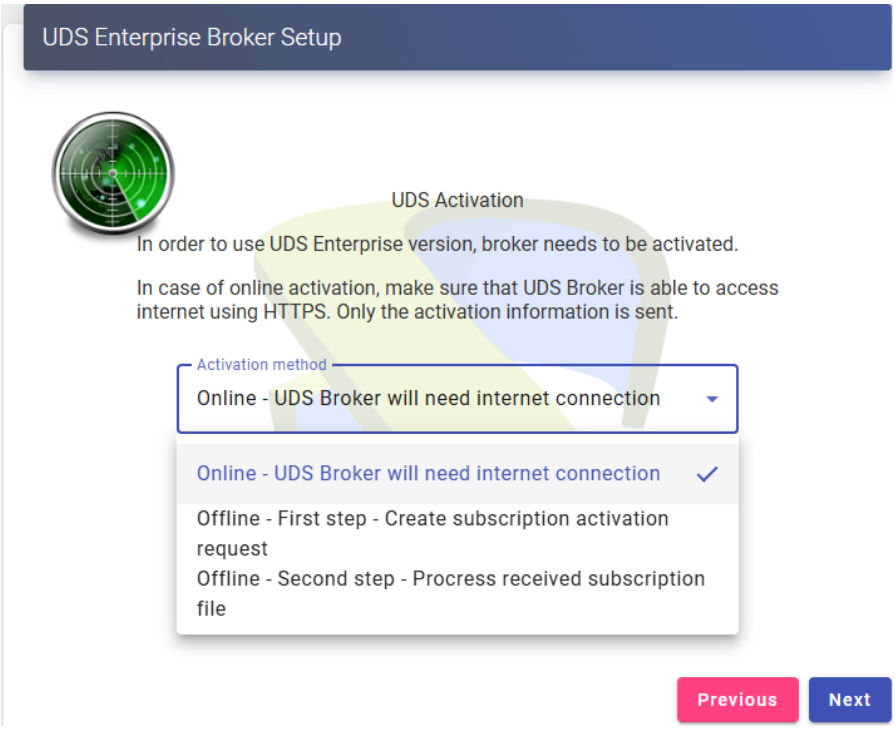
Security Procedure: It is advisable to change the username and password of the database, this is done on the database server itself. Passwords must be of sufficient length and include upper and lower case, numbers and special characters...

Wait for the connection to the database to be set up and continue:



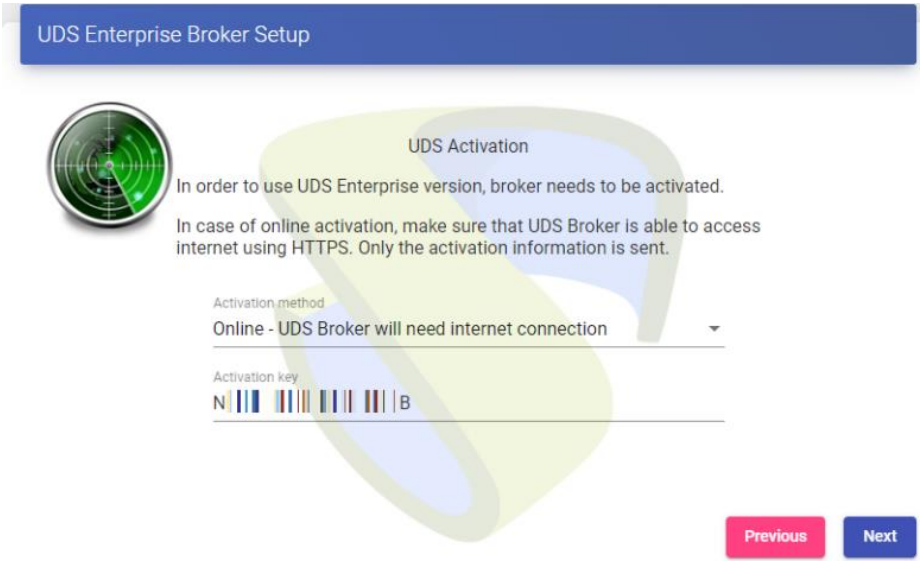
Step 6.- Continue with the subscription activation process.

We must select the activation mode:

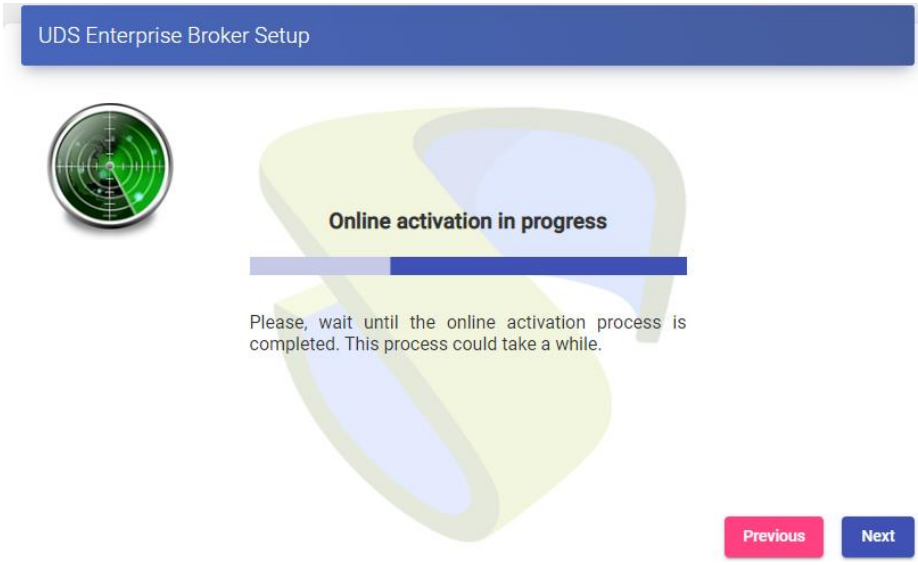


- **Online:** To perform this activation mode, you will need to have a valid serial number and connection to the UDS activation servers on the internet.

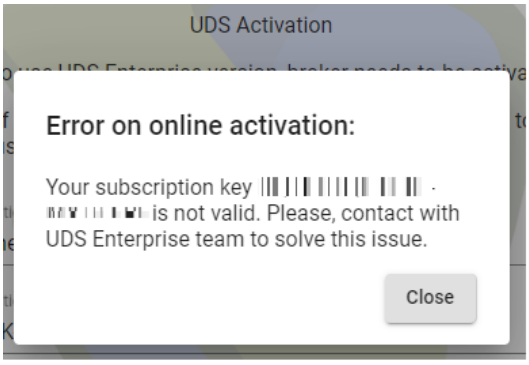
Select the option ***"Online - UDS Broker will need internet connection"*** and enter a valid activation code:

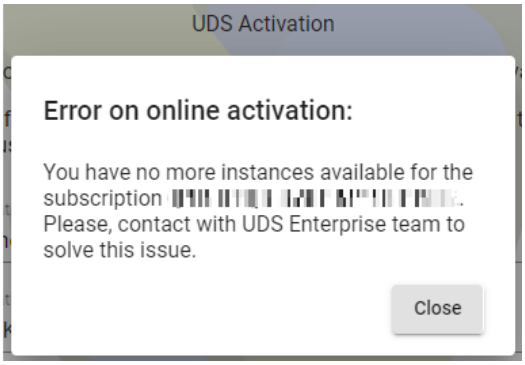


The system will validate the activation code with the UDS central remote servers and, if valid, will continue the configuration process.



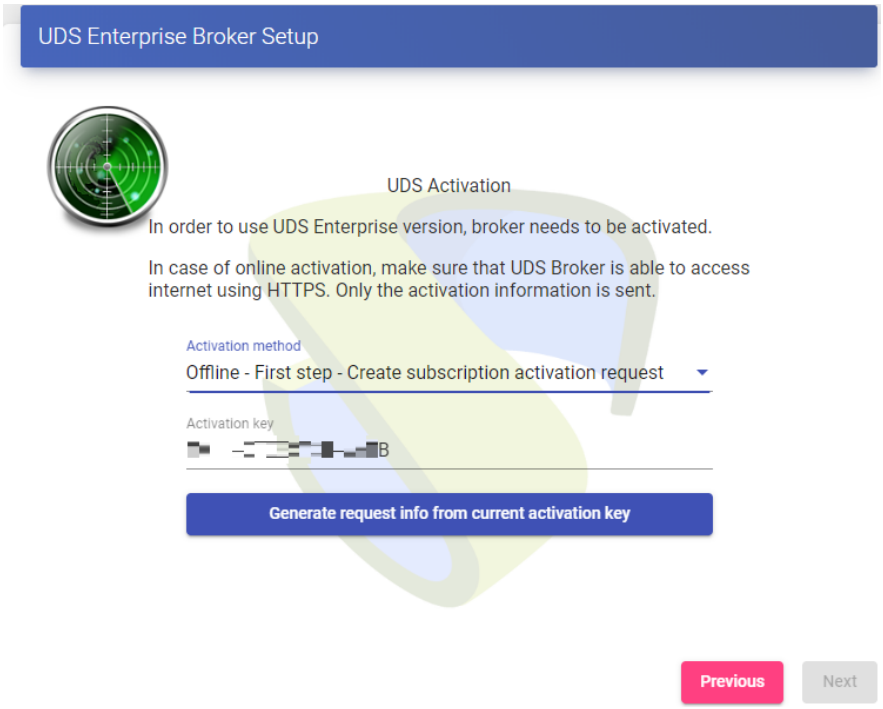
If the activation code is invalid or you do not have more instances, an error will appear in the activation and you will need to contact the UDS Enterprise support team:



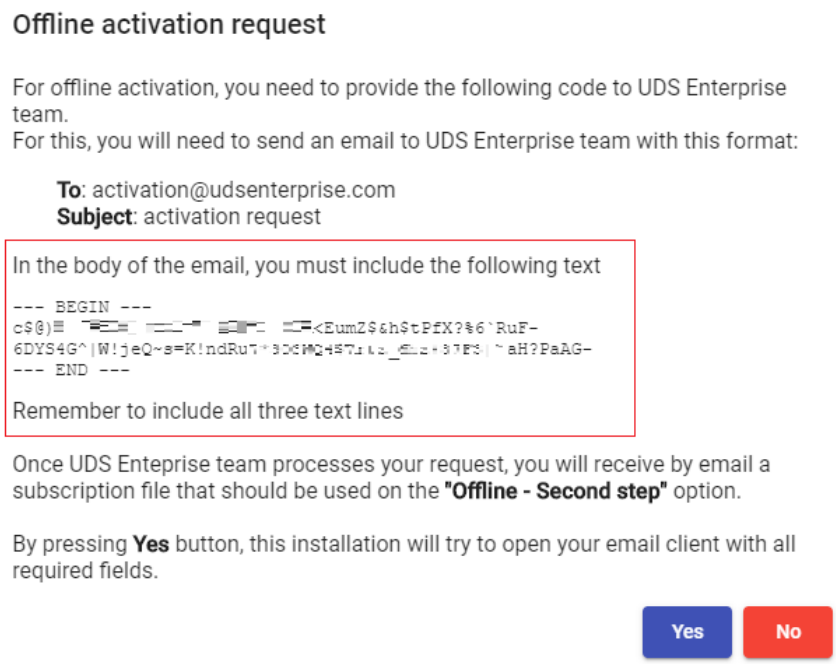


- **Offline:** This subscription activation procedure will only have to be carried out when the UDS server does not have a connection to the UDS activation servers on the internet.

First select the option ***"Offline – First step – Create subscription activation request",*** enter your activation code and click on ***"Generate request info from current activation key":***



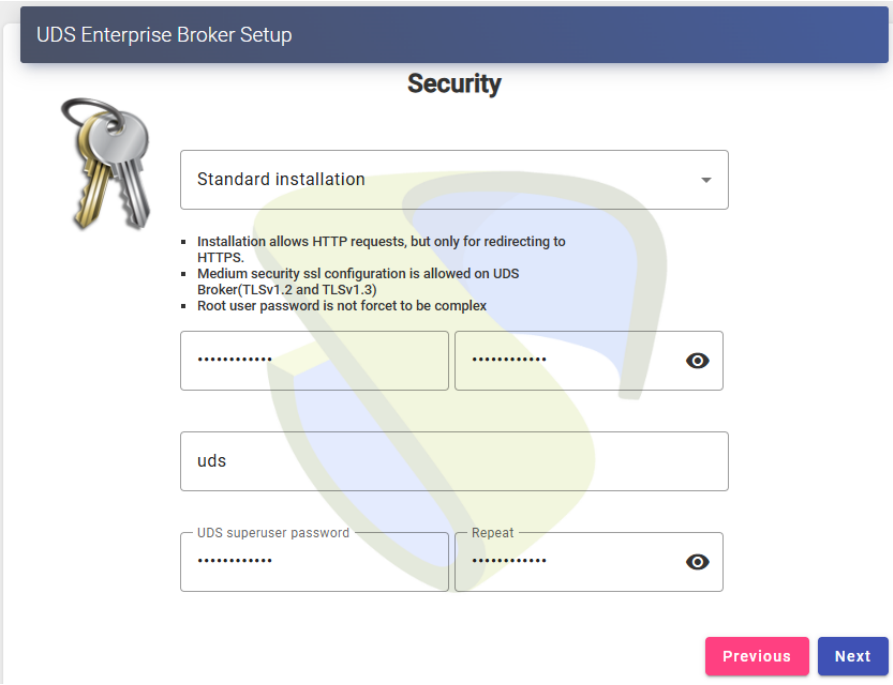
A new window will automatically open with instructions to perform the first part of the activation. It will tell us that we must send, via email, a text automatically generated by the system:



Once we have received the response from the UDS Enterprise support team (which may take some time, since this request has to be processed and validated), we will receive a file called ***uds-subscription.key.***

Now select the option ***"Offline – Second step – Process received subscription file"***, indicating the file received in ***"Current subscription file"*** and continue with the configuration wizard.

Step 7.- Select the security level of the environment, configure the password of the local root user of the UDS server and indicate the name and password of the UDS system administrator user (super-user to access the UDS web administration).



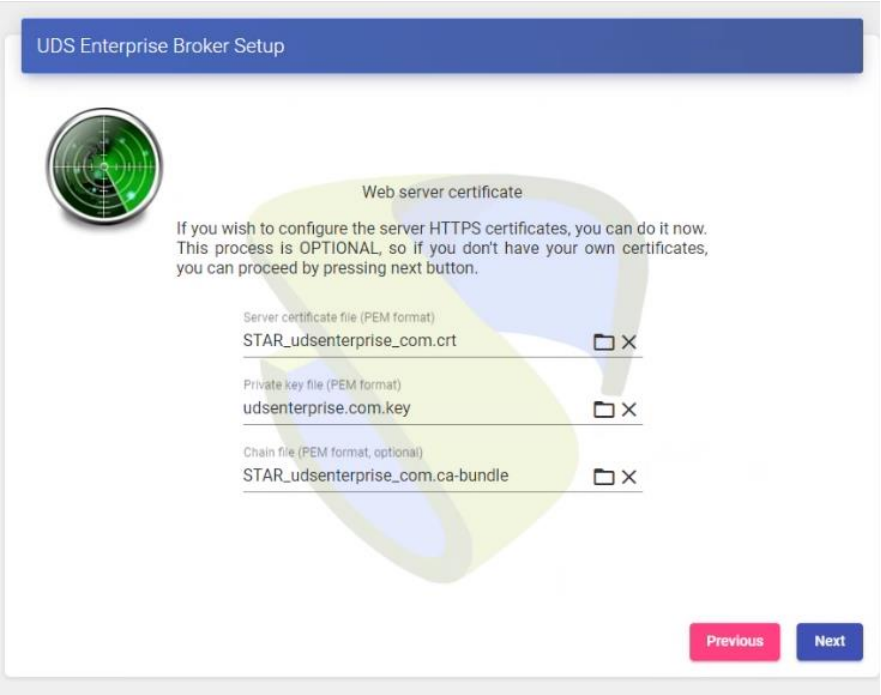
If we choose the "Standard" security level, any request directed to port 80 will be automatically sent to 443, only TLS levels 1.2 and 1.3 classified as secure will be allowed and no complexity will be required in passwords.

If we choose the "Hardenend" security level, no request can be made via port 80, only TLS level 1.3 will be allowed and complexity will be required for passwords.

Procedimiento de Seguridad: Para cualquier nivel de seguridad elegido, se recomienda utilizar contraseñas con un mínimo de 12 caracteres, mayúsculas, minúsculas, números y caracteres especiales.

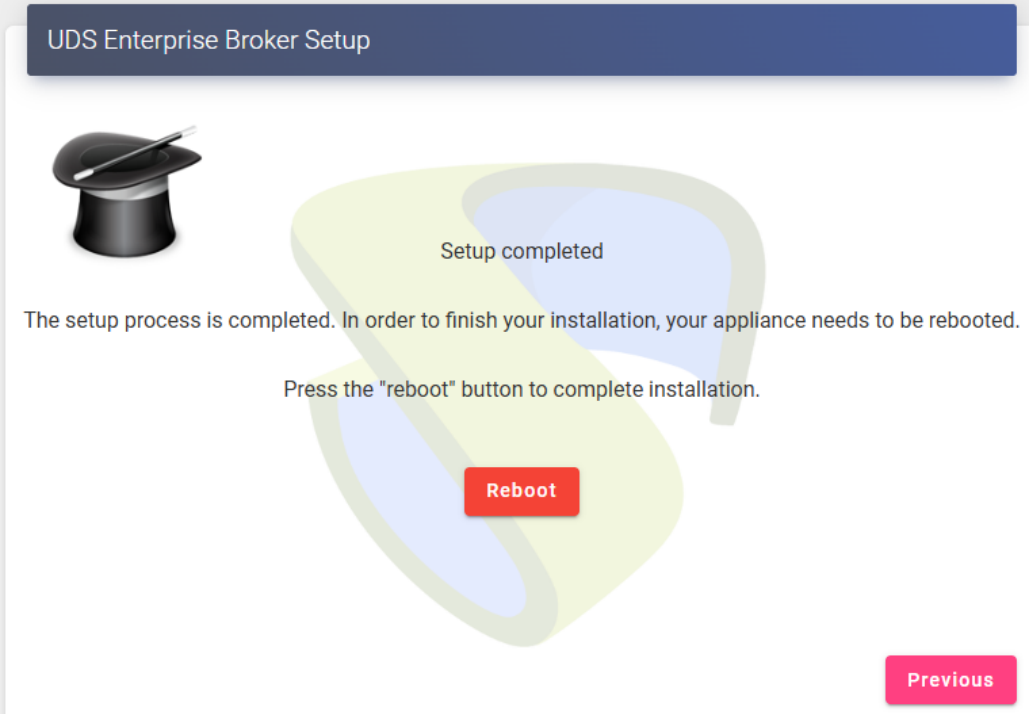
Step 8.- If we have the certificate files, we will indicate them. Otherwise, self-signed ones will be generated and later we can install them via console or even by running the configuration wizard again.

We will need to indicate the certificates in PEM format, the server certificate file in the "Server certificate" field (.crt, .pem, etc...), the file with the key in: "Private key" (.key, .pem, etc...) and optionally we can indicate the chain file of the certification authority "Chain file" (.crt, .pem, etc…).

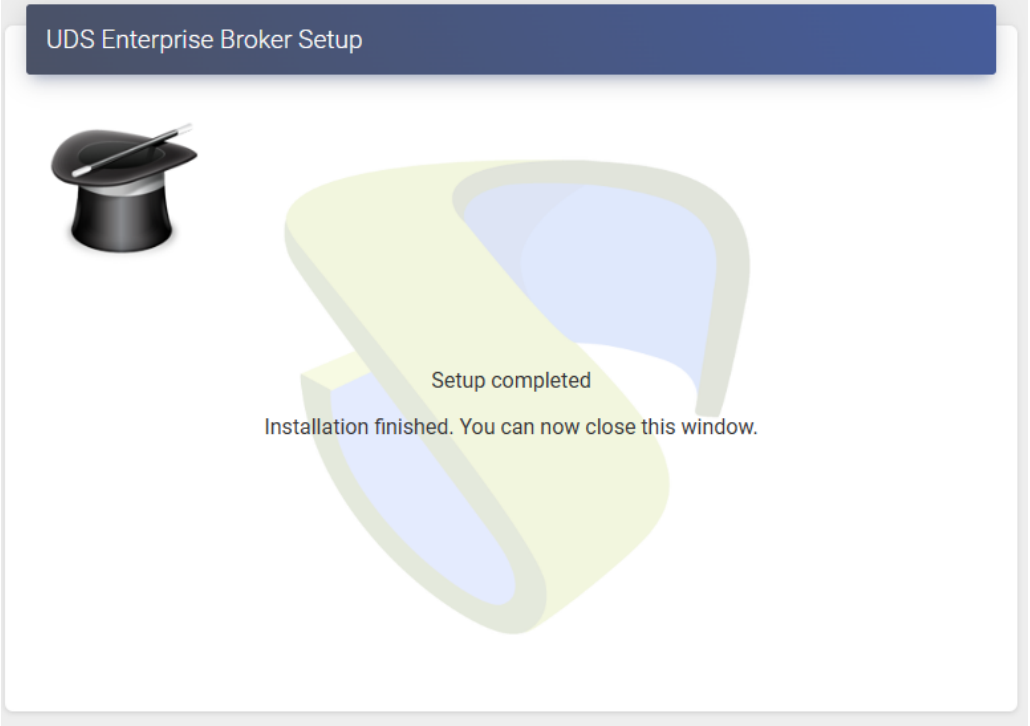


NOTE: For a totally secure installation, and to be able to make connections via HTTPS, the use of valid certificates on the server will always be recommended, otherwise anyone who wants to access the web portal will be warned with an error warning that the self-signed certificates of the UDS Server are insecure.

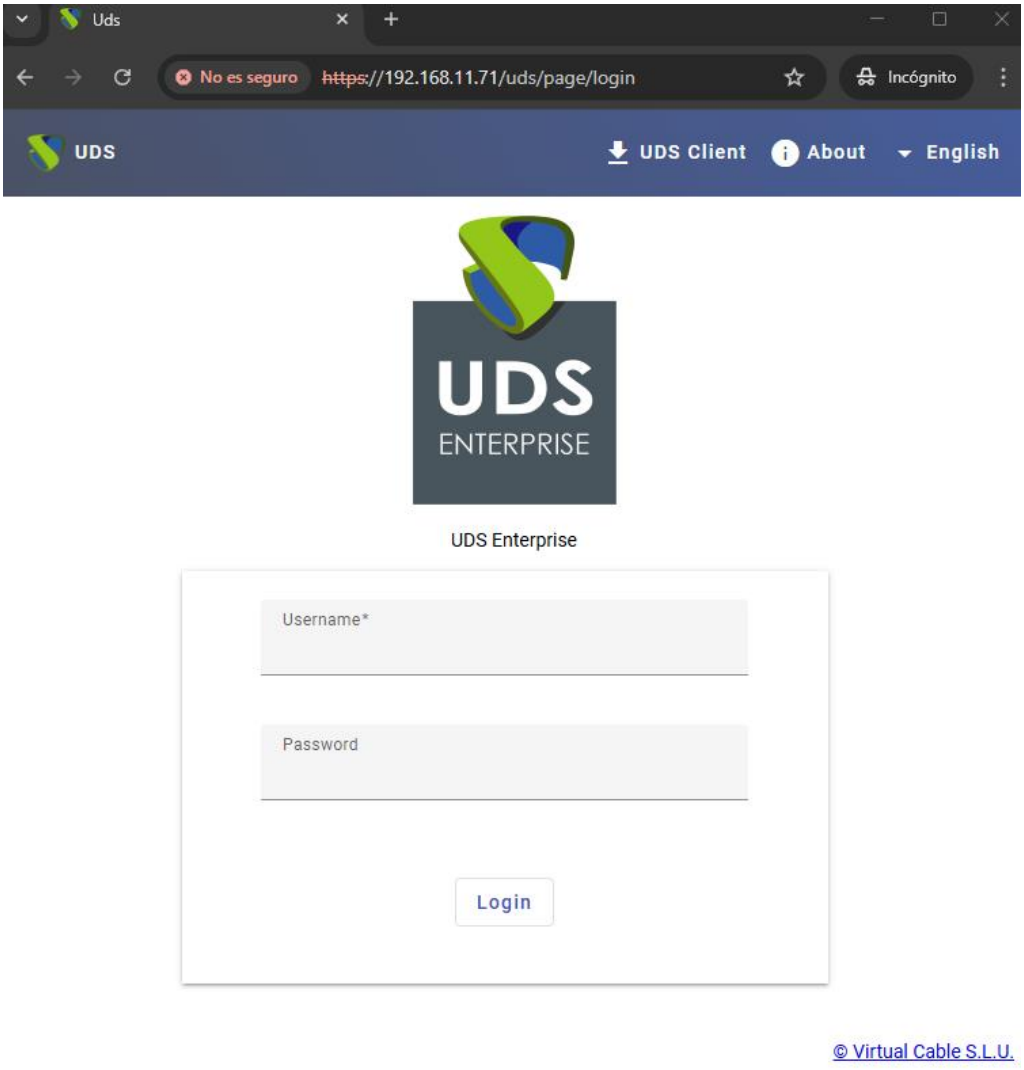
Step 9.- Finish the configuration of the UDS server by clicking on "***Reboot***" so that the server restarts and applies all the indicated configuration.



You will be able to close the configuration wizard page and, once the server has restarted, it will be accessible from any browser by accessing via https the IP address or name of the server:



Home page of UDS Enterprise:

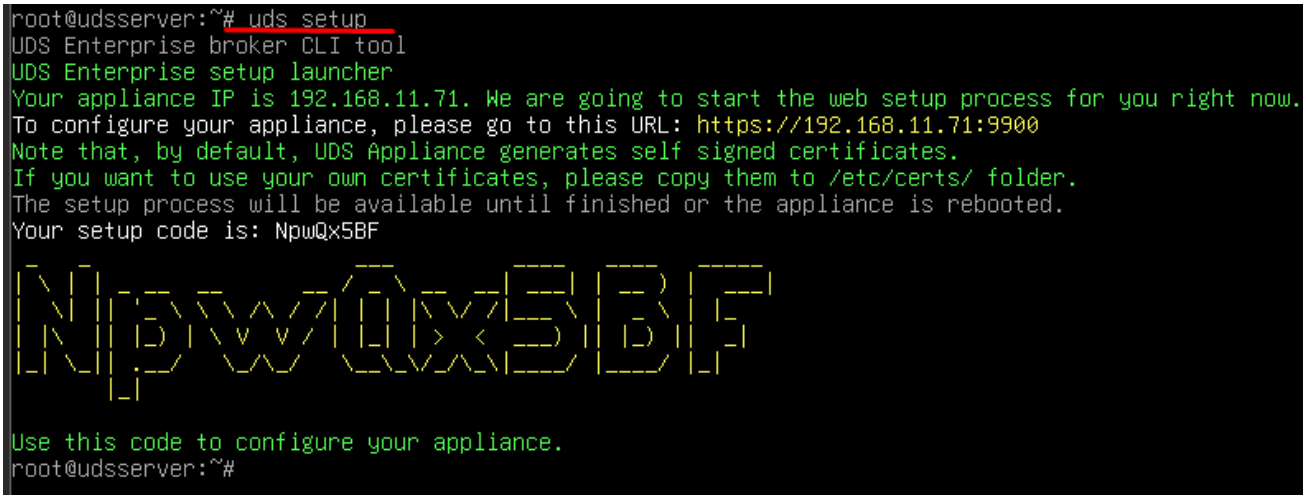


NOTE: Access must be made via HTTPS.

Security procedure: A valid certificate must have been installed on the server in step 8 of the initial basic installation procedure to proceed with the use of UDS Enterprise securely via HTTPS.

If we need to run the configuration wizard again to modify any data, we will have to validate ourselves on the server (with the root user and the password set in step 7) and run the command again:

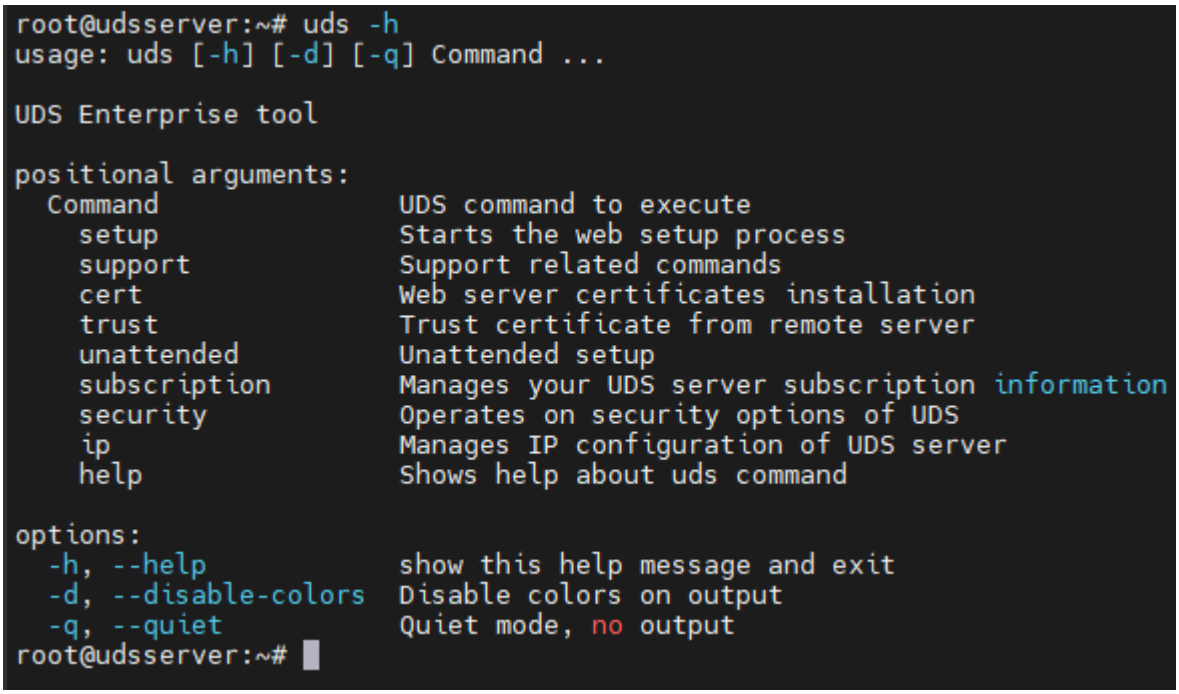
uds setup: Launch the setup wizard**.**



This will enable the setup wizard again. If we need to stop it, we can run the command:

- **uds setup -s**: Detiene el asistente de configuración**.**

Additionally, with the ***uds*** command we can make other modifications to the server:



- **uds support:** It will allow the creation of the support package with all the system configuration. This package will need to be generated when a support request is made.

```
root@udsserver:~# uds support -h
usage: uds support [-h] Action ...

positional arguments:
  Action      Subscription action
    create    Creates the support request bundle.

options:
  -h, --help  show this help message and exit
root@udsserver:~#
```

When the "uds support create*" command is executed*, a .udsbundle file will be generated in the /tmp path that will need to be sent to the UDS Enterprise support team.

```
Processing file /var/log/nginx/error.log...
Processing file /var/log/nginx/error.log.1...
Support file generated at /tmp/uds-support-broker-20250129-183513.udsbundle
root@udsserver:~#
```

- **uds cert:** Allow the installation of certificates on the server:

```
root@udsserver:~# uds cert -h
usage: uds cert [-h] [-c SERVER-CHAIN.PEM] [SERVER-CERT.PEM] [SERVER-KEY.PEM]

positional arguments:
  SERVER-CERT.PEM       Server certificate in PEM format.
  SERVER-KEY.PEM        Server private key in PEM format.

options:
  -h, --help            show this help message and exit
  -c SERVER-CHAIN.PEM, --chain SERVER-CHAIN.PEM
                        Server chain in PEM format.
root@udsserver:~#
```

Podremos incluir el fichero de cadena de la entidad certificadora con el parámetro -c

In addition, as we can see in the following example, when the installation of certificates is executed through this command, the system verifies that the certificate is in valid format and that the indicated key file belongs to that same certificate (when certificates are included through the server's graphical web configuration wizard, no check is performed).

```
root@udsserver:~# uds cert /tmp/Server_cert.pem /tmp/Server_key.pem -c /tmp/ca-bundle.pem
UDS Enterprise broker CLI tool
Reading key file file...done
Reading chain file...done
Checking certificate...Installing certificate...done
root@udsserver:~#
```

- **uds trust:** Command that will allow us to trust a certificate from a remote server.

```
root@udsserver:~# uds trust -h
usage: uds trust [-h] [-n] HOSTNAME PORT

positional arguments:
  HOSTNAME              Hostname of the remote server.
  PORT                  Port of the remote server.

options:
  -h, --help            show this help message and exit
  -n, --no-intermediate
                        Skip intermediate db check (no internet access).
root@udsserver:~#
```

Generally, this command, on this server, is only used to trust self-signed certificates or unrecognized certificate authorities. Additionally, it will add an entry in the /etc/hosts file with the IP address and name of the certificate that has returned the query.

- **uds subscription:** This command performs operations related to the subscription information associated with the serial number.

```
root@udsserver:~# uds subscription -h
usage: uds subscription [-h] Action ...

positional arguments:
  Action      Subscription action
    refresh   Refreshes the subscription information ONLINE (needs internet connection to UDS
              Enterprise servers)
    status    Shows information about your current subscription
    import    Imports a subscription support information file (for renewing offline)

options:
  -h, --help  show this help message and exit
root@udsserver:~#
```

Through its different commands we can:

- **uds subscription refresh**: Consult the UDS activation servers and update the data of the subscription used (it is necessary to have an internet output or at least to the activation servers: keyserver1.udsenterprise.com, keyserver2.udsenterprise.com and keyserver3.udsenterprise.com).
- **uds subscription status**: Displays the information for the current subscription used. This information is hosted in the database instance, if there is no connectivity it will not be able to be viewed.
- **uds subscription import**: Used for the activation/renewal of a subscription when the "online" mode cannot be used. It will be necessary to indicate the activation file provided by the UDS Enterprise team.

- **uds security:** It will allow you to modify the name and password of the super-user, created in the server configuration wizard, we can also enable or disable said user for certain addresses or IP ranges. This user will allow access to the UDS web administration without the need to validate against any authenticator.

```
root@udsserver:~# uds security -h
usage: uds security [-h] Element ...

positional arguments:
  Element     Security operation
    username  Changes the username of admin UDS user
    password  Changes the password of admin UDS user
    allow     Sets allowed admin IP access

options:
  -h, --help  show this help message and exit
root@udsserver:~#
```

  - **uds security username:** Modify the name of the UDS super-user administrator.
  - **uds security password:** Modifica la contraseña del super-usuario administrador de UDS.

Safe use procedure: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.

  - **uds security allow**: Allows you to choose the ips/range that will be able to access the administration.

- **uds ip:** It will allow you to query, modify and reset the IP data and server name:

```
root@udsserver:~# uds ip --help
usage: uds ip [-h] Action ...

positional arguments:
  Action      IP configuration action
    get       Displays the current server IP configuration
    set       Sets the IP configuration of UDS server
    reset     Resets the IP configuration of UDS server to default (DHCP)

optional arguments:
  -h, --help  show this help message and exit
root@udsserver:~#
```

  - **uds ip get**: Displays the current IP and server name settings.
  - **uds ip set**: Permite configurar datos IP y nombre del servidor.
  - **uds ip reset:** Resets the server's network configuration to the initial state (DHCP).

Safe Employment Procedure:

In general, it is advisable to implement basic and essential security mechanisms on any server; strong passwords, backups, having security solutions, keeping systems updated and modifying the configurations, usernames and passwords included, by default.

In addition, as it is a web server, Nginx, it is recommended to apply the following mechanisms:

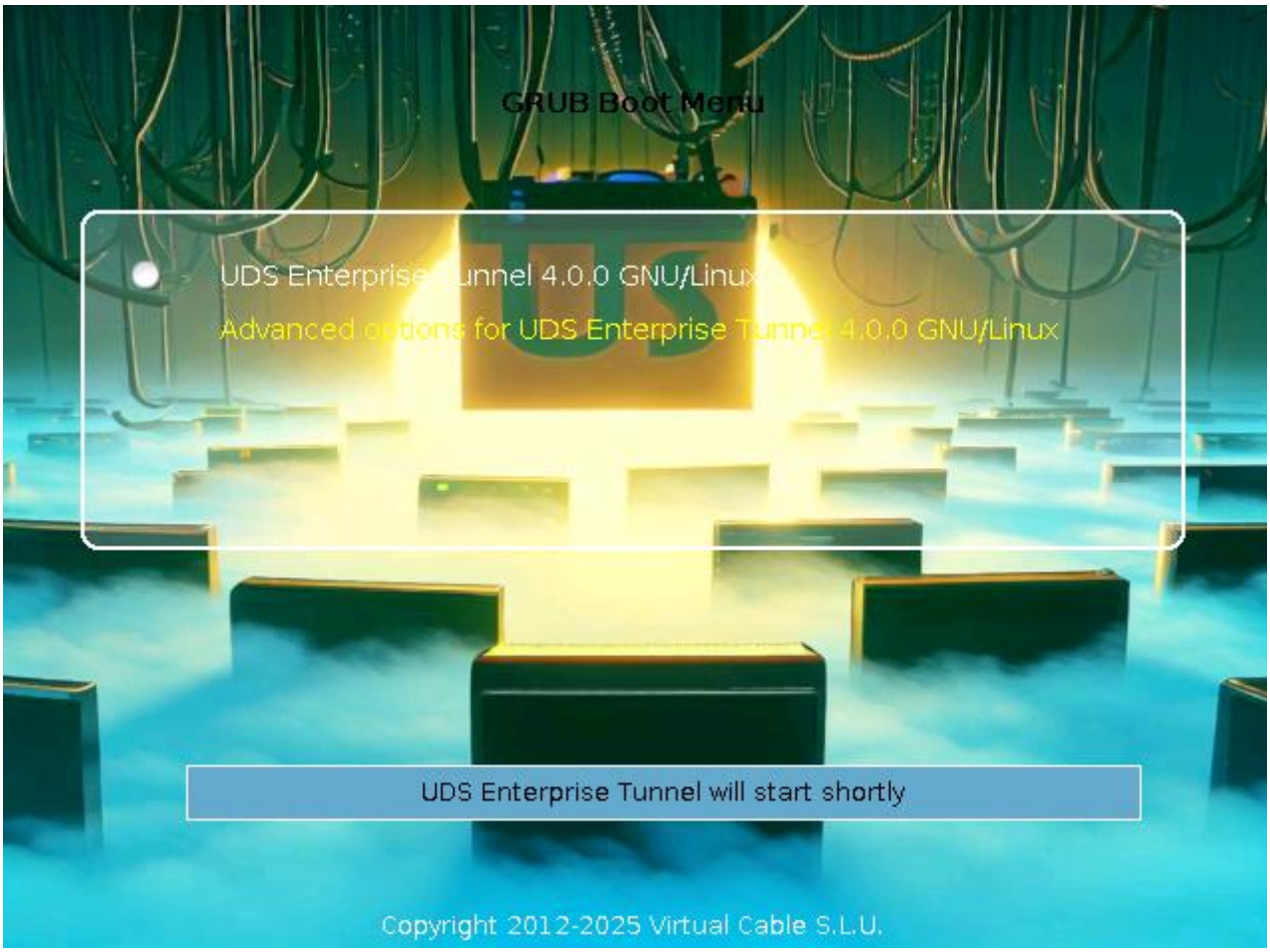
- SSL/TLS
    - Implement SSL Certificate
    - Optimize SSL / TLS
    - Disable weak SSL/TLS protocols
    - Disable weak encryption suites
    - Install Chain Certificate

  Securing Diffie-Hellman for TLS
- Erase “Information Leakage”
- Web Application Security
    - Disable unwanted HTTP methods
    - Prevent clickjacking attacks, x-frame injection
    - X-XSS Protection
    - Implement WAF Mod Security

It is advisable to disable SSH access to this server, so that it is only accessible by console.

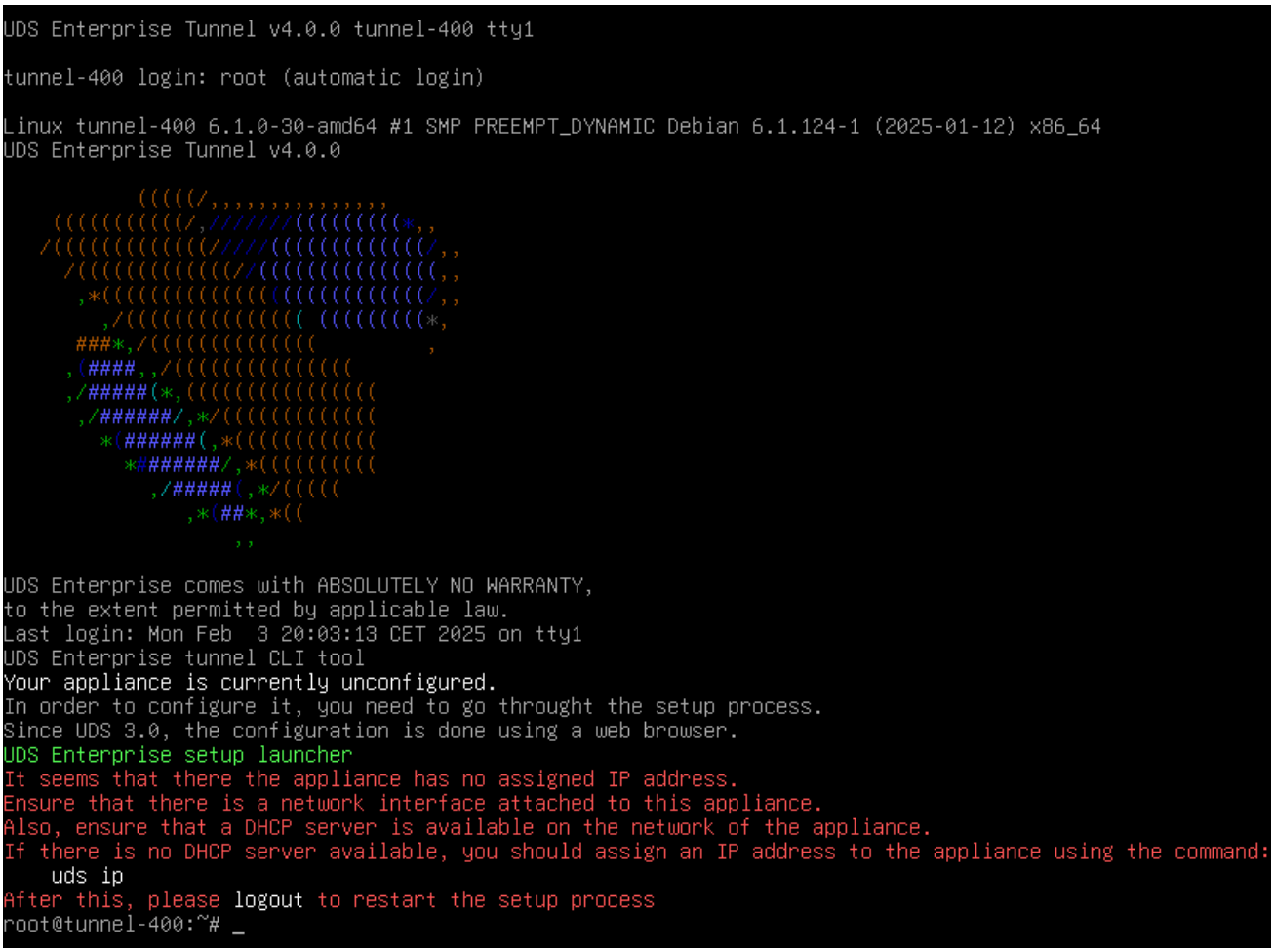
3.2.3 UDS Tunnel

Once the UDS Tunnel virtual appliance has been imported into a supported virtualization platform, we turn on the virtual machine to proceed with its initial configuration.

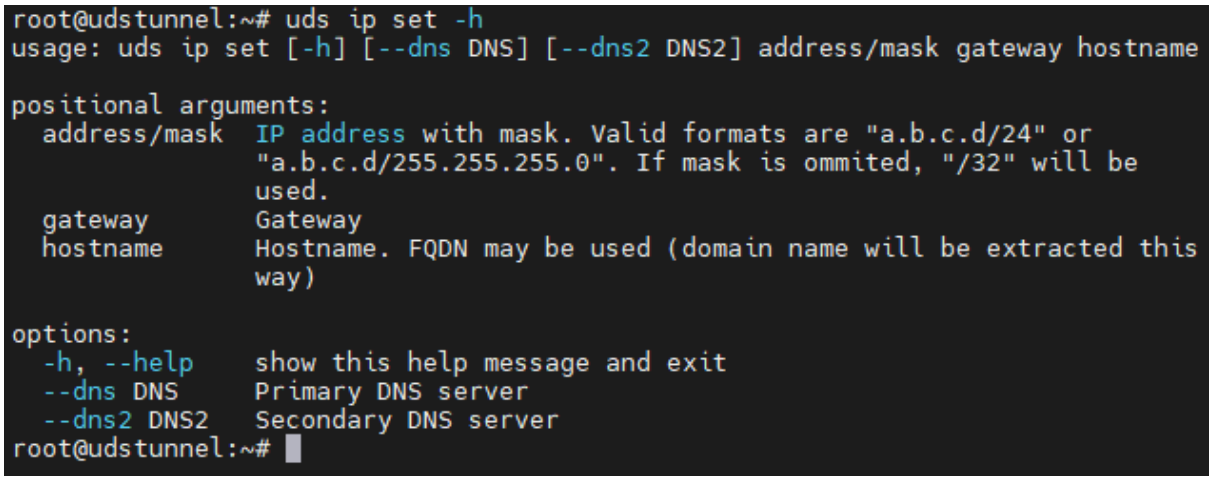


NOTE: In order to successfully configure a UDS Tunnel server, it is necessary to have a UDS server previously configured and to know its IP address or name.

By default, the UDS Tunnel virtual appliance will take a network configuration via DHCP. In case there is no server on the network that assigns IP addresses, we will have to assign the network data manually:



To do this we will use the command: ***uds ip set*** with the configuration options:



Proceed with the manual configuration of the server's network data:

```
uds ip set ip_server/mask gateway name_server
```

Additionally, we can indicate the domain (extracted from the server name) and the DNS servers (with the --dns parameter)

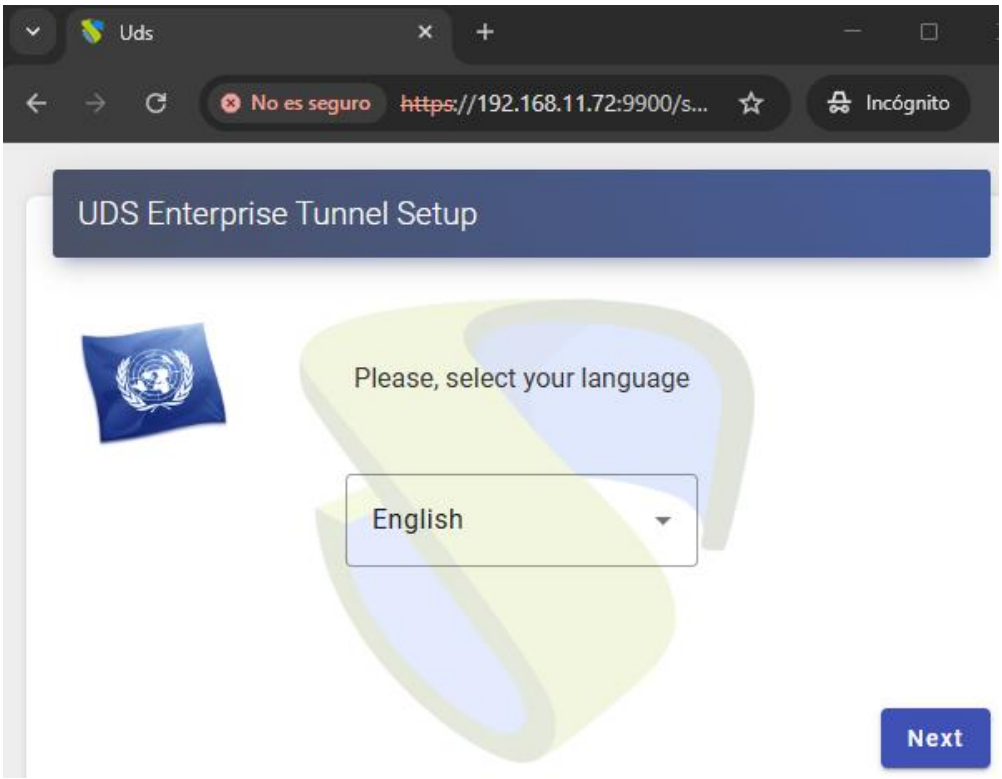
```
root@tunnel-400:~# uds ip set 192.168.11.72/24 192.168.11.1 udstunnel.vc.local --dns 192.168.11.1
UDS Enterprise tunnel CLI tool
Updating network configuration...[  718.204884] e1000: eth0 NIC Link is Up 1000 Mbps Full Duplex,
[  718.205613] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
done
New network configuration
DHCP: no
Using interface: eth0
Hostname: udstunnel
Domain: vc.local
Address: 192.168.11.72
Mask: 255.255.255.0
Gateway: 192.168.11.1
DNS: 192.168.11.1
Secondary DNS: 80.58.61.250
You need to reboot your appliance in order to fully activate the new configuration
root@tunnel-400:~#
```

Once the IP data is configured, restart the server to apply the changes.

If we already have an IP address assigned to the server, either by its manual configuration or by the automatic assignment of a DHCP server, we will proceed to the configuration of the UDS Tunnel component.

To do this, access via web browser (**with https**) the IP address of the server with port 9900:

```
UDS Enterprise setup launcher
Your appliance IP is 192.168.11.72. We are going to start the web setup process for you right now.
To configure your appliance, please go to this URL: https://192.168.11.72:9900
Note that, by default, UDS Appliance generates self signed certificates.
```



Safety note:

To carry out the initial basic configuration, UDS Tunnel incorporates its own security mechanism.

Para poder realizar la configuración inicial necesitamos dos cosas:

- The "uds setup" command must be running in the broker
- We need to know the value of our server's Setup Code.

The "uds setup" command is executed automatically the first time the server is turned on and can be executed manually at any time from the console.

The "uds setup" command stops automatically when the initial setup is finished or manually by running "uds setup -s".

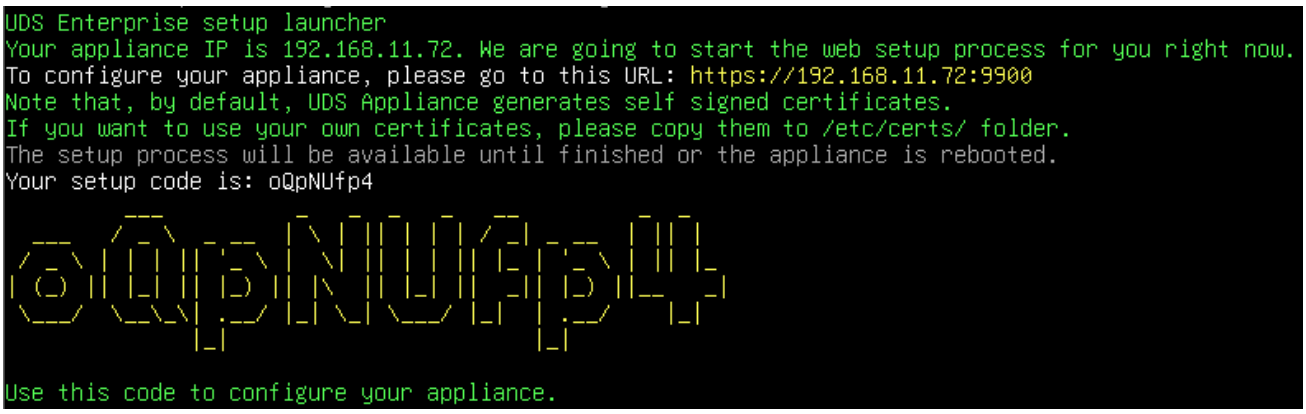
When executing the "uds setup" command, the "Setup Code" of our server appears on the screen, which will be requested if we want to perform the basic configuration.

The "Setup Code" is an eight-character token that **does NOT** circulate at any time on the network, which will be our "One-time pad".

"One-time pad" is part of an encryption technique that cannot be decrypted, but requires the use of a pre-shared, one-time key that is no smaller than the message being sent.

In this technique, plain text is combined with a random secret key, our one-time "One-time pad".

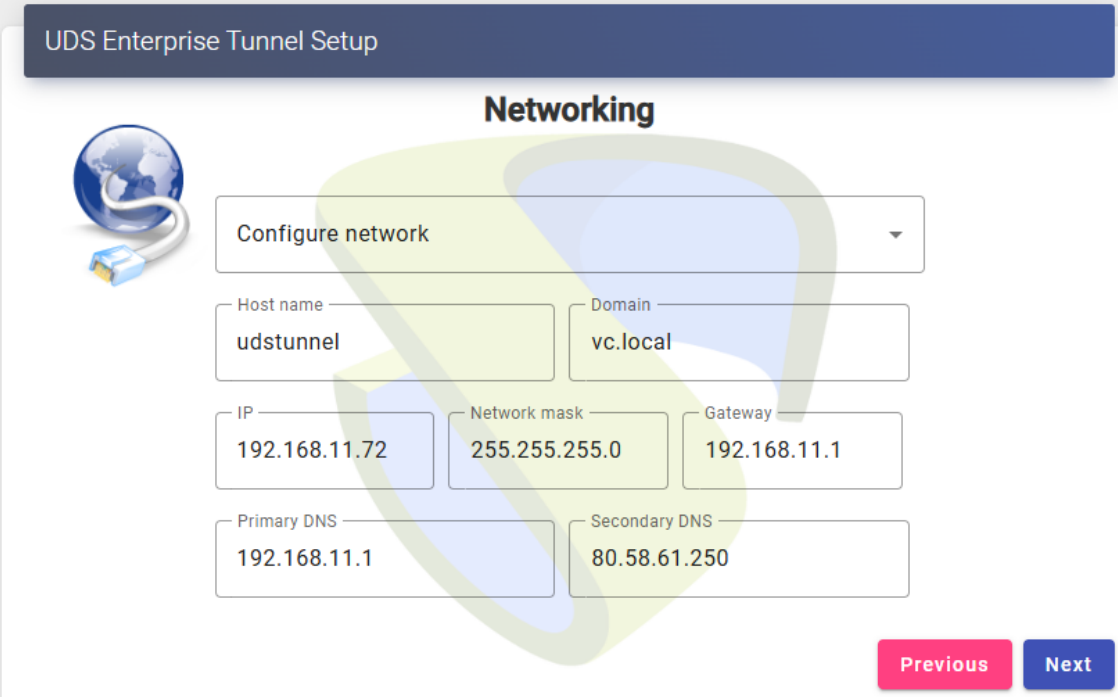
With this token at each end of the connection using a "salt", a set of random bits that we use as part of the cryptographic algorithm key, and using a 512 hash function, we obtain a 3,072-byte key, which is chopped up and used by both parties to encrypt and decrypt the information transmitted and received during this initial configuration. This key is never used in its entirety or reused in any of its parts.



This "Setup Code" will be requested during the initial basic configuration.

Step 1.- Select the language of the configuration wizard:

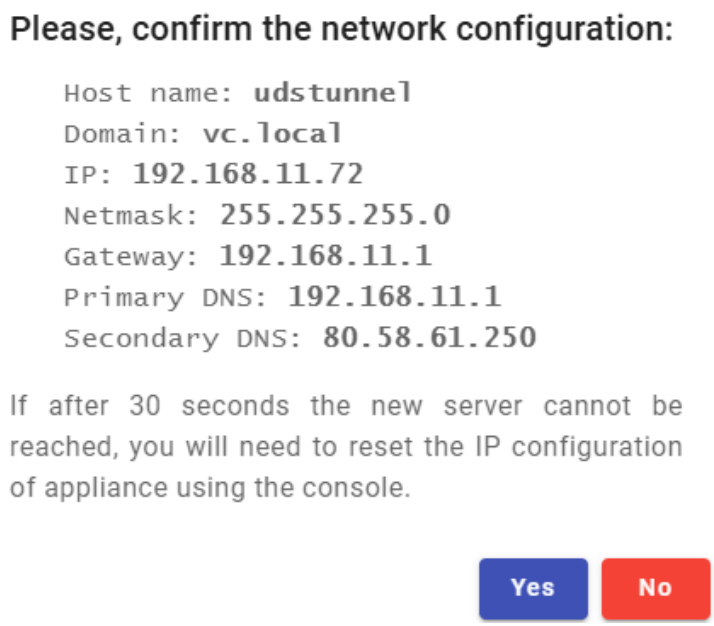
Step 2.- Indicate the server name, domain (optional) and network data of the server:



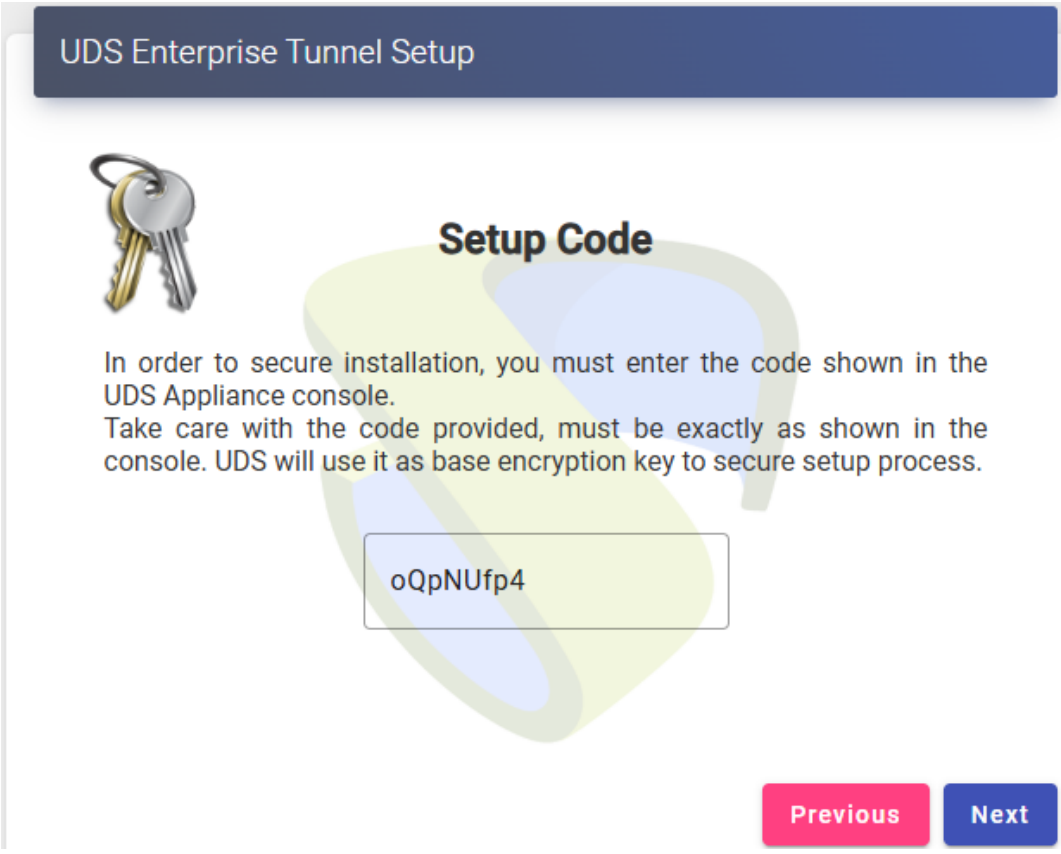
NOTE: If the IP address indicated is different from the one already configured by the server via DHCP or via manual configuration, the system will automatically redirect to the new address to continue with the configuration wizard.

NOTE: If all the data indicated is correct and you do not want to modify any data, you can use the option *"Skip network config (leave it as is)"*

Check that the data indicated are correct and we accept:

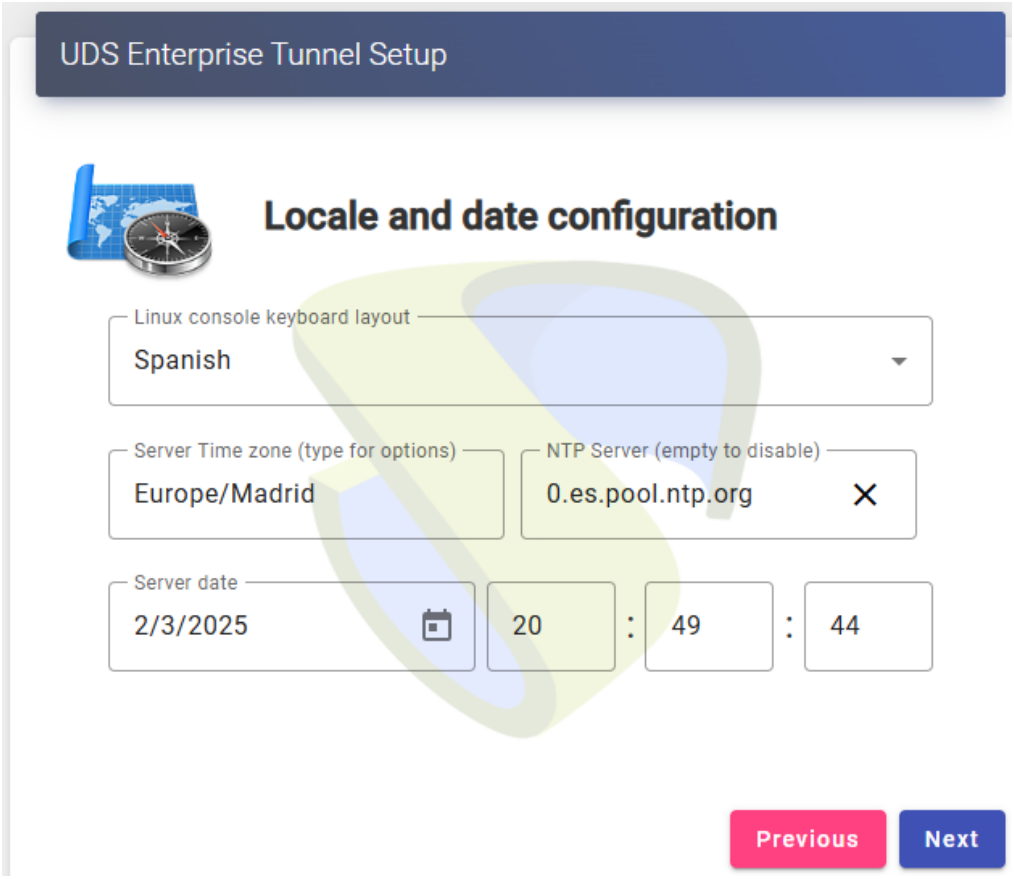


Step 3.- Added the security code ("Setup Code") that will appear in the console of our UDS Tunnel appliance and that we saw at the beginning of this procedure:



If you do not have access to the server console, via ssh you can locate the code in the file: /etc/setupcode.uds

Step 4.- Configure the keyboard language that the server will have, the time zone, and an NTP server (optional):



It is very important to select the time zone correctly, otherwise, there may be problems with certain functionalities (MFAs, authentication via SAML, etc...) and the system events will not show the time correctly.

Step 5.- We register the Tunnel server with the UDS server (Broker), for this we indicate the type of connection (from UDS version 3.6 it has to be HTTPS), the name of the UDS server (as it has to be through a secure connection, the IP address cannot be used), we select an authenticator and a user with administration permissions of the selected authenticator.

If we have not installed any certificate on the UDS server and it still maintains the self-signed certificate by default, in order for the tunnel server to trust this self-signed certificate, it will be necessary to launch the "uds trust" command on the tunnel server:

```
root@udstunnel:~# uds trust -h
usage: uds trust [-h] [-n] HOSTNAME PORT

positional arguments:
  HOSTNAME              Hostname of the remote server.
  PORT                  Port of the remote server.

options:
  -h, --help            show this help message and exit
  -n, --no-intermediate
                        Skip intermediate db check (no internet access).
root@udstunnel:~# _
```

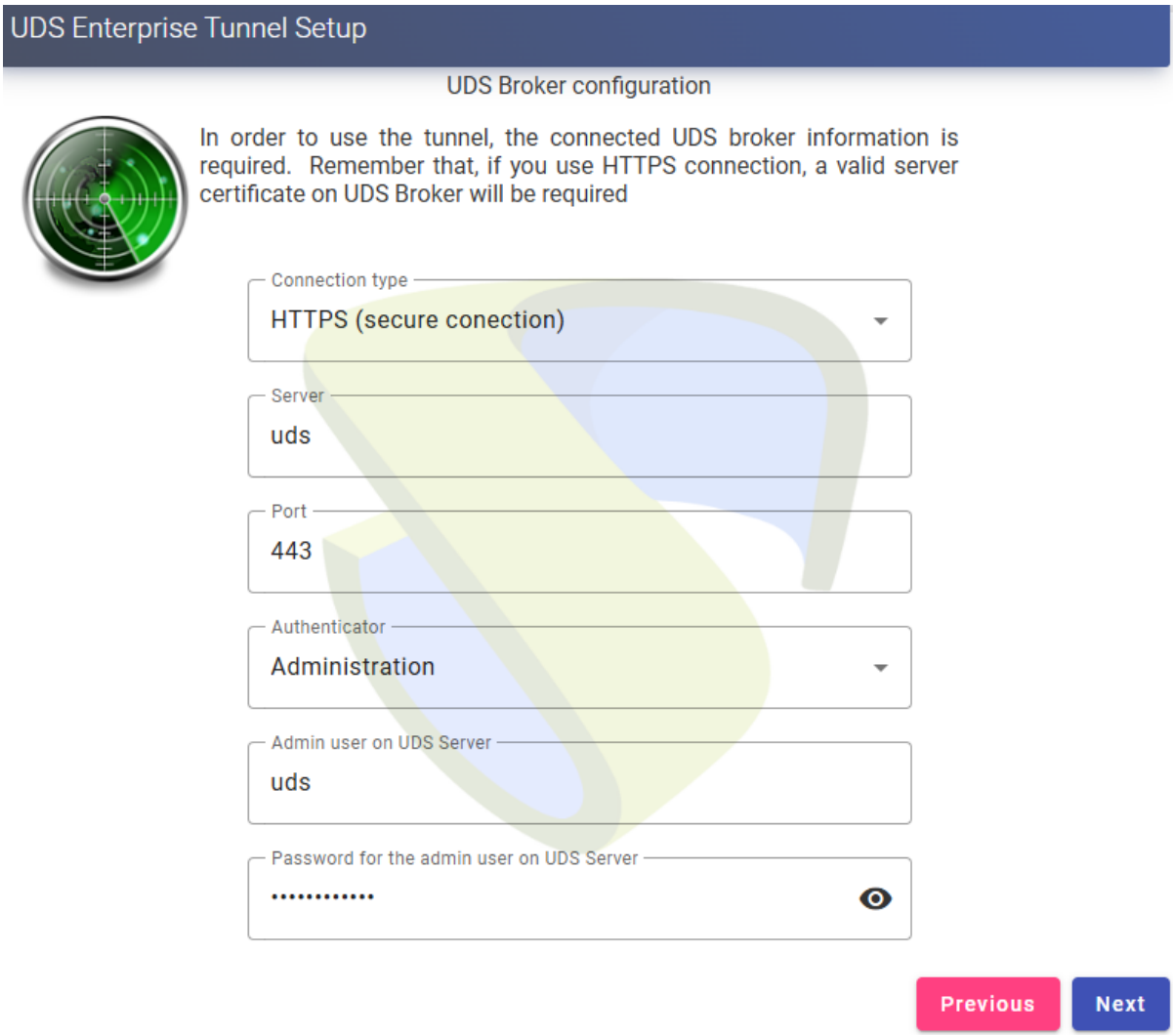
As indicated in the help, we will specify the IP address or name with the port, which in this case will be 443:

```
root@udstunnel:~# uds trust 192.168.11.71 443
UDS Enterprise tunnel CLI tool
Reading certificate from server 192.168.11.71:443 done
Certificate name: uds
Valid from: 2025-01-27 16:48:31+00:00
Valid until: 2035-01-25 16:48:31+00:00
Fingerprint: 19ce23ebccadffc5cadc9bf5cbaf49dba85167fb3654385e477fb8a9615f3059
Issuer: CN=uds,O=UDS Enterprise Self Signed Certificate,L=Madrid,ST=Madrid,C=ES
Subject: CN=uds,O=UDS Enterprise Self Signed Certificate,L=Madrid,ST=Madrid,C=ES
Serial number: 537973254630396632499273476695444437756304847634
Self signed: Yes
Self signed certificate. Trusting it...
Writing certificate to trust file (/usr/local/share/ca-certificates/uds.crt)... done
Ensuring that the name uds resolves to the IP 192.168.11.71...
updating /etc/hosts... done
Updating trusted database...rehash: warning: skipping duplicate certificate in TERENA_
rehash: warning: skipping ca-certificates.crt,it does not contain exactly one certific
done
Trusted certificate installed
root@udstunnel:~#
```

The tool will add the self-signed certificate named "uds" to the list of valid certificates and will also create an entry in the /etc/hosts file so that the resolution of the certificate name is effective:

```
  GNU nano 7.2                                    /etc/hosts
# Autogenerated by UDS installer
127.0.0.1       localhost
127.0.1.1       udstunnel.vc.local udstunnel

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
192.168.11.71   uds
```

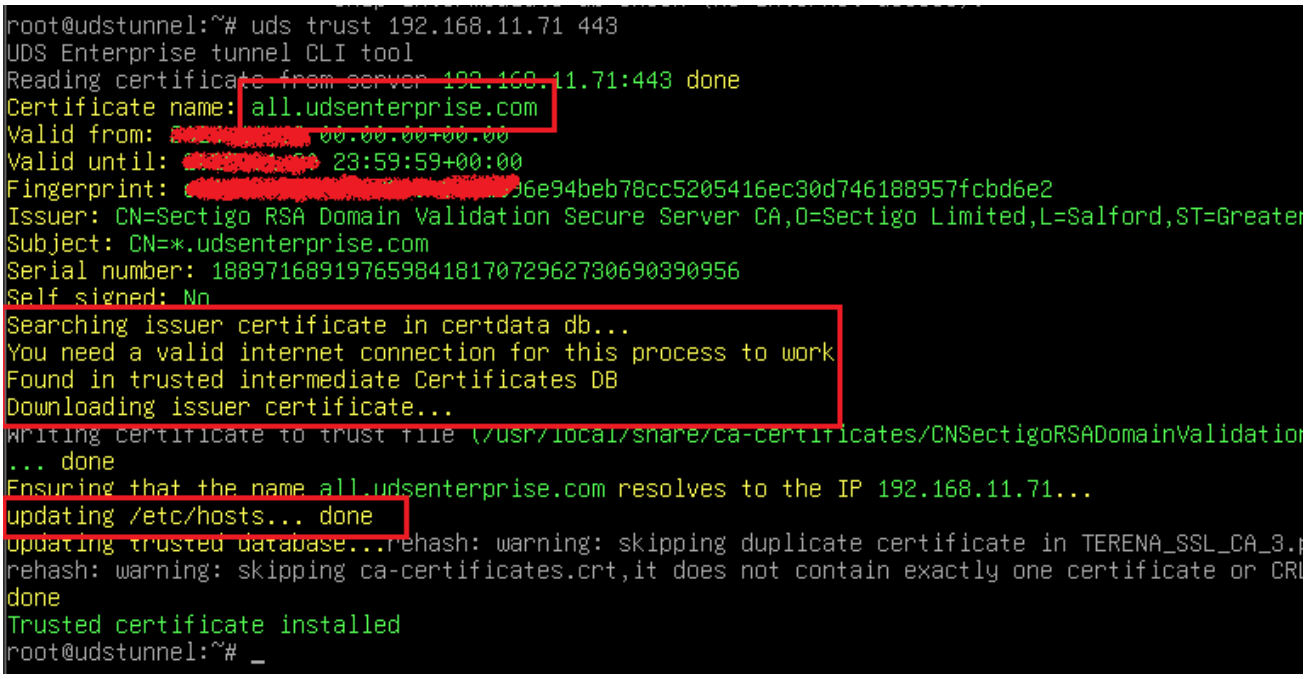
Once these tasks are completed, the UDS server's self-signed certificate will be "trusted" by the Tunnel server, allowing you to proceed with the configuration wizard.



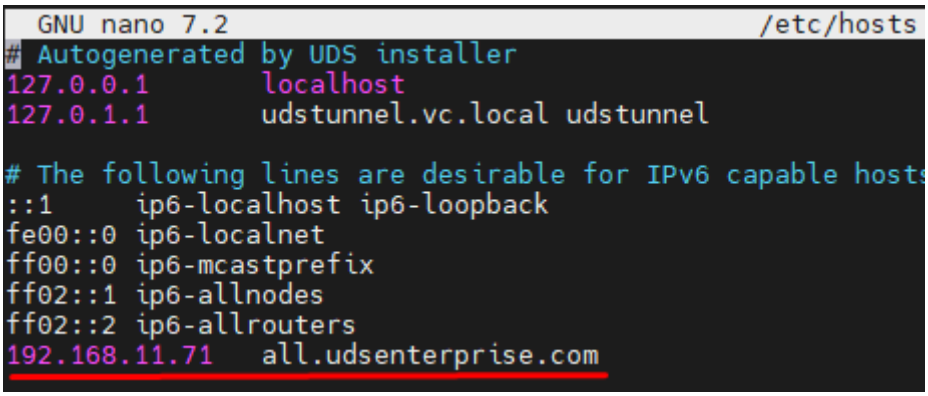
NOTE: If we do not have an authenticator configured on the UDS server, we can use the "Administration" authenticator to which the super-user indicated in step 7 of the UDS server configuration wizard belongs.

If we have installed a valid certificate on the UDS server and recognized by the main certification authorities, it will not be necessary (in principle) to execute the command "uds trust", although if it is executed we will see that it indicates the exact name of the certificate (if it is of the "wildcard" type it will add "all" in the root of the name), it will add the entry to the hosts file and if an intermediate entity is required and we have internet connectivity, It will download it automatically (the same way a web browser works).

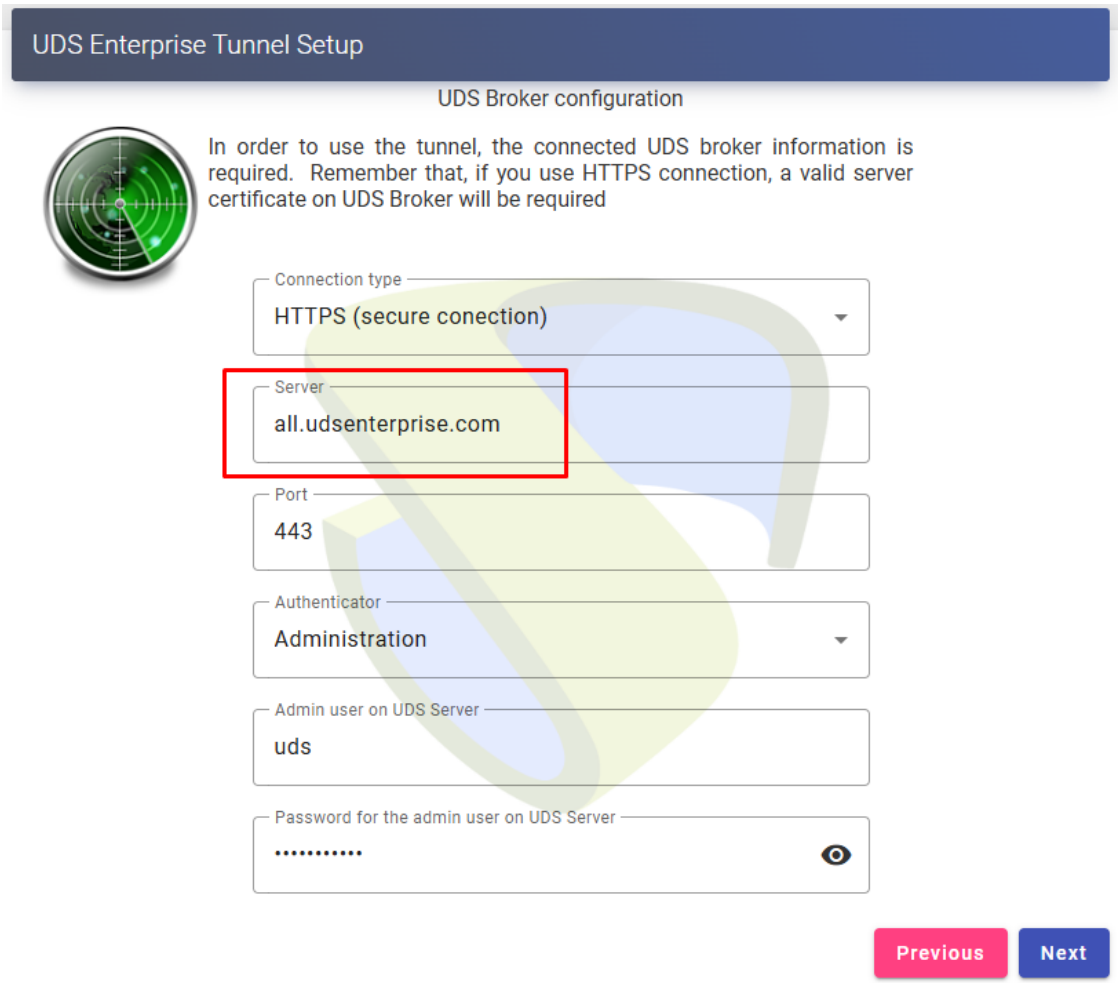
Execution of the "uds trust" command from the Tunnel server to the IP of a UDS server with a "Wildcard" certificate:



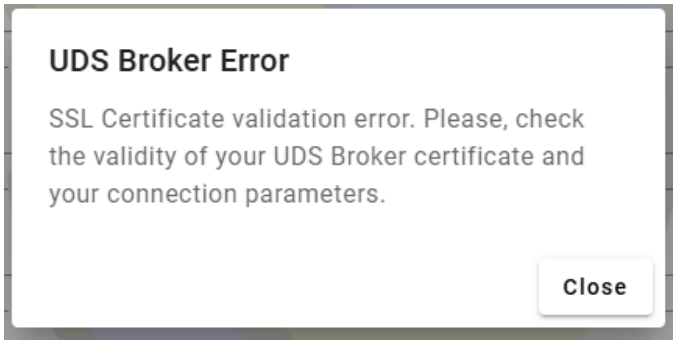
In this case, the entry in the hosts file would look like this:



And in the tunnel registry we would write:



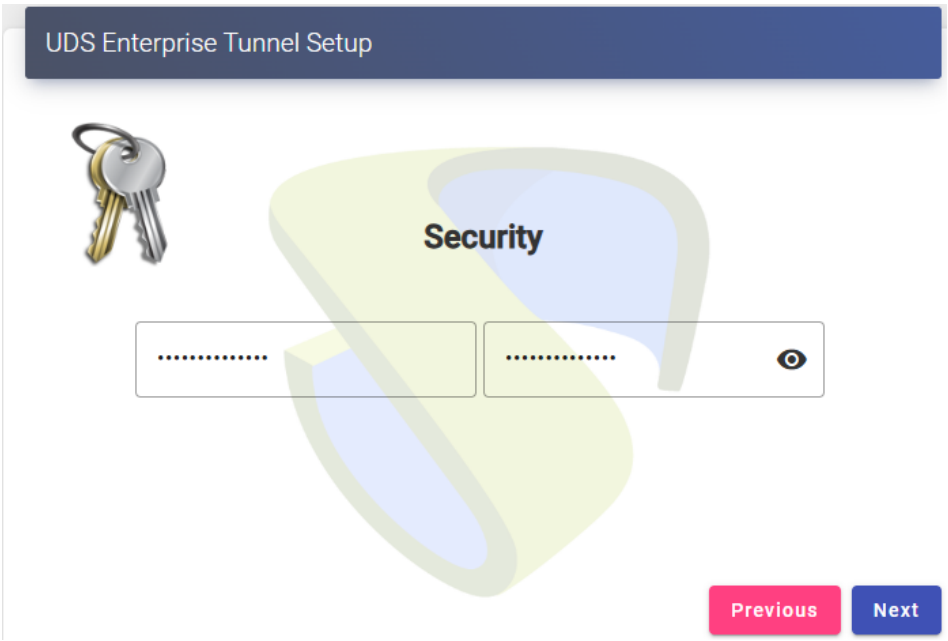
NOTE: If at any time we get an error such as "SSL Certificate validation error":



It may indicate that the certificate installed on the UDS server is not "trusted" by the broker and the following tasks need to be performed on the Tunnel server:

- Run the "uds trust" command with internet connectivity to add the necessary intermediate CAs.
- Manually add the CA to the certificate store because you are using a self-signed or do not have internet connectivity to perform it via "uds trust". In this case we will copy the file containing the entire path of CAs to /usr/local/share/ca-certificates/ and run the command: *update-ca-certificates --fresh*

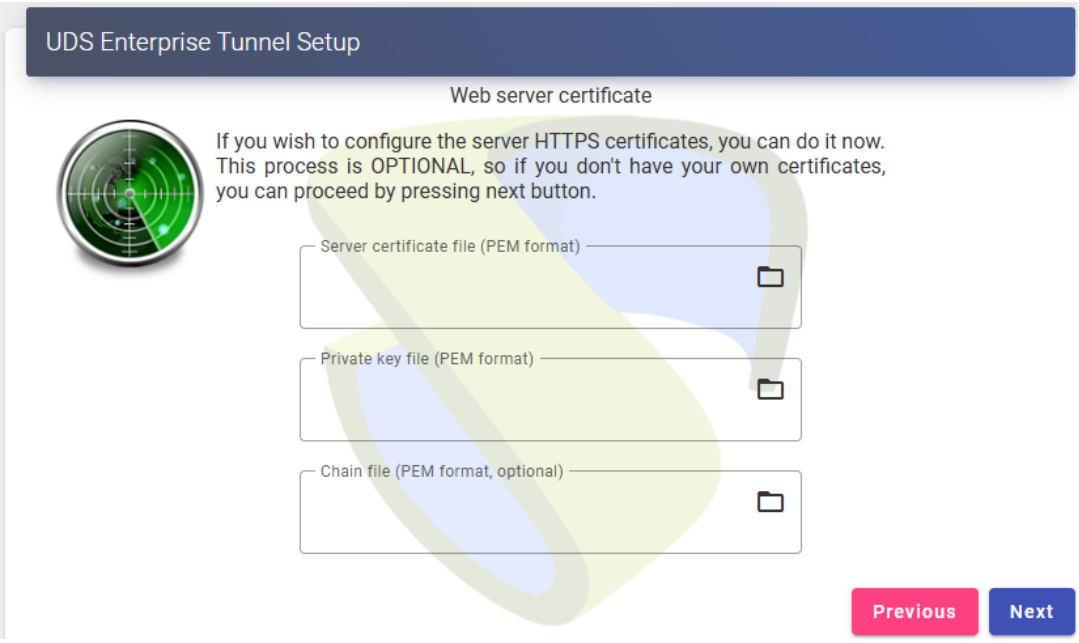
Step 6.- We set the password for the local root user of the Tunnel server:



NOTA: Para una instalación totalmente segura se recomienda utilizar contraseñas con un mínimo de 12 caracteres, mayúsculas, minúsculas, números y caracteres especiales.

Step 7.- If we have the certificate files, we will indicate them. Otherwise, self-signed ones will be generated and later we can install them via console or even by running the configuration wizard again.

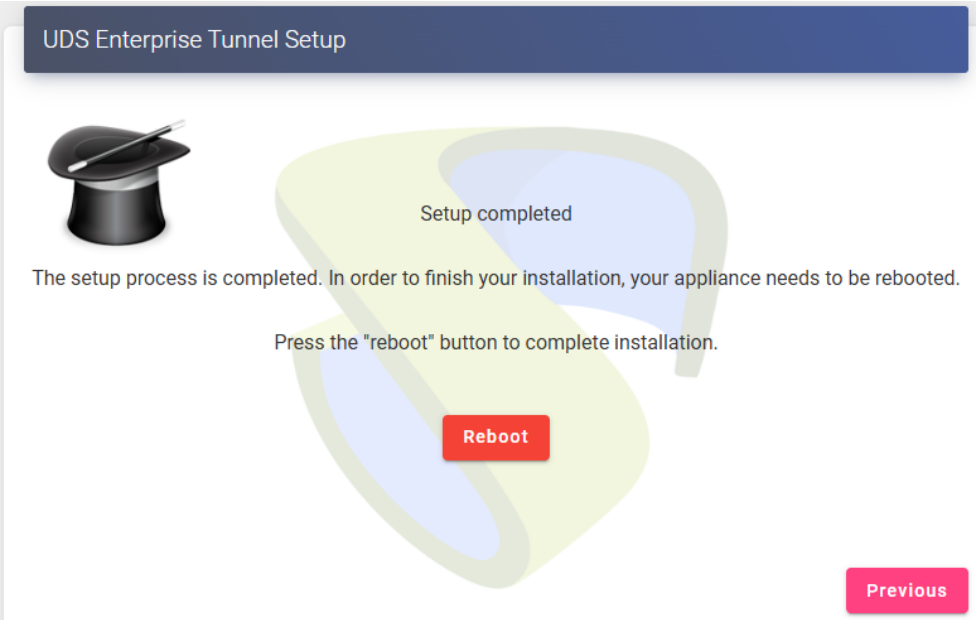
We will need to indicate the certificates in PEM format, the server certificate file in the "Server certificate" field (.crt, .pem, etc...), the file with the key in: "Private key" (.key, .pem, etc...) and optionally we can indicate the chain file of the certifying authority "Chain file" (.crt, .pem, etc...).



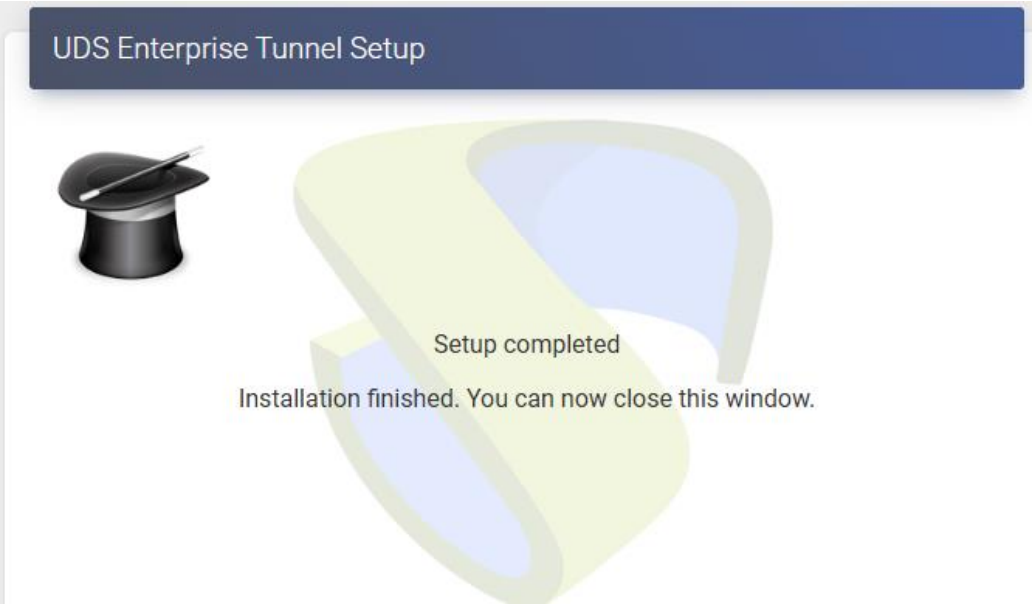
NOTE: This certificate will be used only for HTML5 connection mode, because for tunneled connections, even if they use port 443, it is not an http traffic.

For a totally secure installation, and to be able to make HTML5 connections via HTTPS, the use of valid certificates on the server will always be recommended since otherwise anyone who wants to access with the HTML5 connection will be warned of an error warning that the self-signed certificates of the UDS Tunnel are insecure.

Step 8.- Finish the configuration of the Tunnel server by clicking on "***Reboot***" so that the server restarts and applies all the indicated configuration.

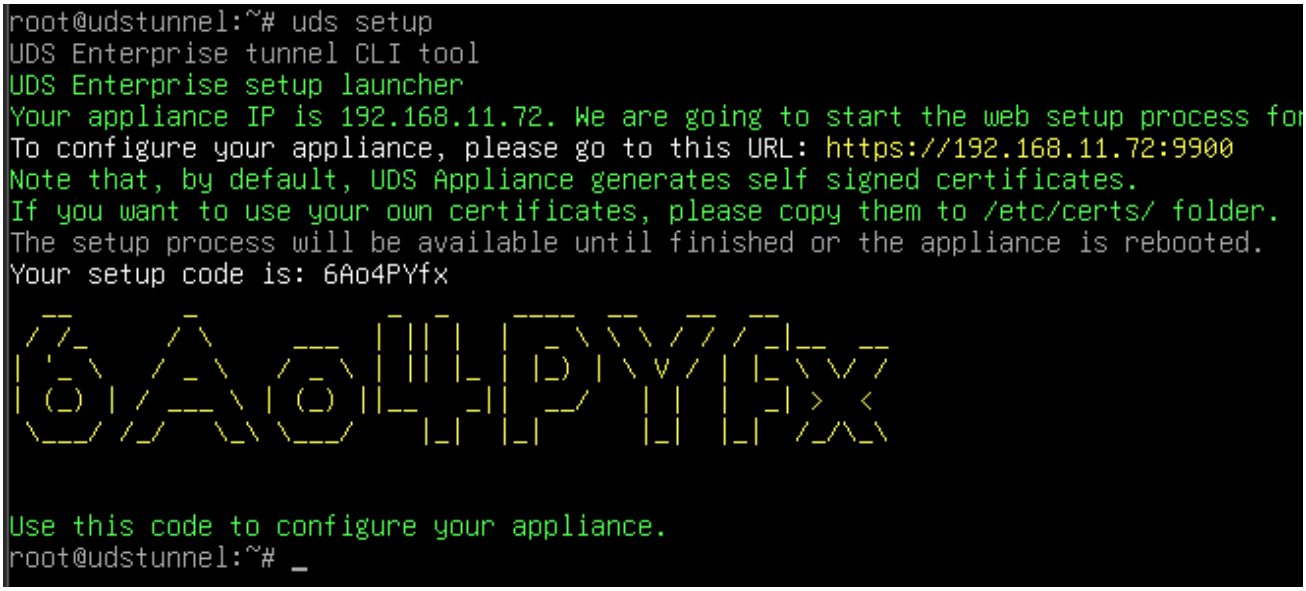


Close the configuration wizard page and, once restarted, the UDS Tunnel server will be fully configured.



If we need to run the configuration wizard again to modify any data, we will have to validate ourselves on the server (with the root user and the password set in step 6) and execute the command:

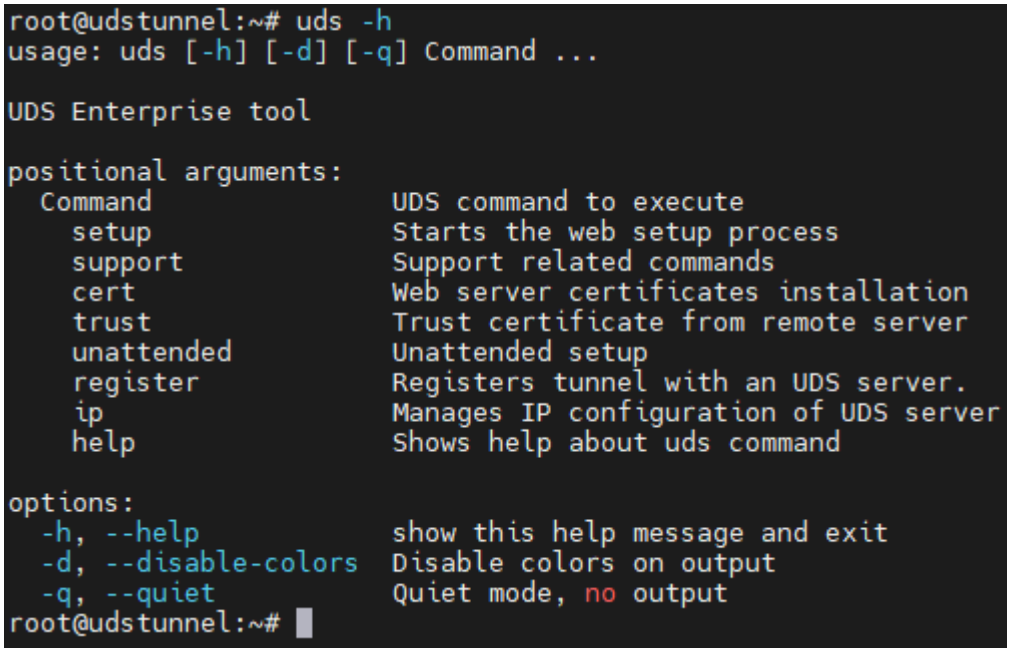
- **uds setup:** Launch the Server Setup Wizard



This will enable the setup wizard again. If we need to stop it, we can run the command:

- **uds setup -s**: Stops the setup wizard**.**

Additionally, with the ***uds*** command we can make other modifications to the server:



- **uds support:** It will allow the creation of the support package with all the system configuration. This package will need to be generated when a support request is made.

```
root@udstunnel:~# uds support -h
usage: uds support [-h] Action ...

positional arguments:
  Action      Subscription action
    create    Creates the support request bundle.

options:
  -h, --help  show this help message and exit
root@udstunnel:~#
```

When the "uds support create*" command is executed*, a **.udsbundle** file will be generated in the /tmp path that will need to be sent to the UDS Enterprise support team.

```
Processing file /var/log/tomcat9/localhost.2025-01-27.log...
Processing file /var/log/tomcat9/localhost.2025-02-04.log...
Support file generated at /tmp/uds-support-tunnel-20250204-173842.udsbundle
root@udstunnel:~#
```

uds cert: It will allow the installation of certificates on the server:

```
root@udstunnel:~# uds cert -h
usage: uds cert [-h] [-c SERVER-CHAIN.PEM] [SERVER-CERT.PEM] [SERVER-KEY.PEM]

positional arguments:
  SERVER-CERT.PEM       Server certificate in PEM format.
  SERVER-KEY.PEM        Server private key in PEM format.

options:
  -h, --help            show this help message and exit
  -c SERVER-CHAIN.PEM, --chain SERVER-CHAIN.PEM
                        Server chain in PEM format.
root@udstunnel:~#
```

We can include the chain file of the certification authority with the -c parameter

In addition, as we can see in the following example, when the installation of certificates is executed through this command, the system verifies that the certificate is in valid format and that the indicated key file belongs to that same certificate (when certificates are included through the server's graphical web configuration wizard, no check is performed).

```
root@udstunnel:~# uds cert /tmp/Server_cert.pem /tmp/Server_key.pem -c /tmp/ca-bundle.pem
UDS Enterprise tunnel CLI tool
Reading key file file...done
Reading chain file...done
Checking certificate...Installing certificate...done
root@udstunnel:~#
```

- **uds trust:** Command that will allow us to trust a certificate from a remote server.

```
root@udstunnel:~# uds trust -h
usage: uds trust [-h] [-n] HOSTNAME PORT

positional arguments:
  HOSTNAME              Hostname of the remote server.
  PORT                  Port of the remote server.

options:
  -h, --help            show this help message and exit
  -n, --no-intermediate
                        Skip intermediate db check (no internet access).
root@udstunnel:~#
```

uds register: It will allow you to register the Tunnel server on a specific UDS server (broker). The process is the same as that carried out during the configuration wizard (step 5), requiring the same data.

```
root@udstunnel:~# uds register -h
usage: uds register [-h] [-s] [server] [auth] [username]

positional arguments:
  server      UDS Server host[:port]. Use this alone to get the authenticators list.
  auth        Authenticator auth name or uuid
  username    UDS Server auth username (must be admin)

options:
  -h, --help  show this help message and exit
  -s, --ssl
root@udstunnel:~#
```

We will also have to execute this command when we modify the certificate of the UDS server, to get its certificate to be "trusted" again through the tunnel. For example, if we have kept the UDS server with the self-signed certificate and registered the tunnel with that certificate, when we apply the final certificate to the UDS server, we will have to re-register the Tunnel.

In the following example of tunnel logging with a UDS server (broker), we first launch the command:

`uds register` *`name_UDSserver`*`:443 -s`

```
root@udstunnel:~# uds register all.udsenterprise.com:443 -s
UDS Enterprise tunnel CLI tool
No authenticator selected. Listing:
UUID                                 Name           Label
------------------------------------ -------------- --------------
5326f891-5cb8-5916-a8ab-efa832edea2a AD             ad
00000000-0000-0000-0000-000000000000 Administration Administration
c46697c2-aee3-5d4a-aadd-81a51be17a98 Internal       int
root@udstunnel:~#
```

This command will return all the authenticators available on the UDS server, we must choose one of them and use a user with administration permissions (the "Administration" authenticator would be the one that includes the super-user created in the UDS server configuration wizard, step 7).

Once the authenticator has been chosen and with an administrator user of it, we will execute the command:

```
uds register name_UDSserver:443 authenticator username -s
```

```
root@udstunnel:~# uds register all.udsenterprise.com:443 Administration uds -s
UDS Enterprise tunnel CLI tool
Using Administration authenticator (label Administration)
Please, enter the password for uds:
Trying to register with UDS Server on all.udsenterprise.com:443 using authenticator Administration and user uds wi
th the provided credentials...
root@udstunnel:~#
```

NOTE: It is necessary to indicate the name of the server and port 443, the IP address and port 80 cannot be used. In addition, the UDS server must have a certificate recognized by the tunnel server (as was done in the configuration wizard, it is possible to previously use the "uds trust" command).

- **uds ip:** It will allow you to consult, modify and reset the data IP of the Server:

```
root@udstunnel:~# uds ip -h
usage: uds ip [-h] Action ...

positional arguments:
  Action      IP configuration action
    get       Displays the current server IP configuration
    set       Sets the IP configuration of UDS server
    reset     Resets the IP configuration of UDS server to default (DHCP)

options:
  -h, --help  show this help message and exit
root@udstunnel:~#
```

  - **uds ip get**: Displays the current IP and server name settings.
  - **uds ip set**: Allows you to configure IP data and server name.
  - **uds ip reset:** Resets the server's network configuration to the initial state (DHCP).

Safe Employment Procedure:

In general, it is advisable to implement basic and essential security mechanisms on any server; strong passwords, backups, having security solutions, keeping systems up to date and modifying the configurations, usernames and passwords included, by default.

In addition, since it includes a web server, Nginx, it is recommended to apply the following mechanisms:

- SSL/TLS
  - Implement certificate SSL
  - Optimize SSL / TLS
  - Disable weak protocols SSL / TLS
  - Disable encryption weak suites
  - Install Certificate string
  - Securing Diffie-Hellman for TLS
- Eliminate “Information Leakage”
- Web Application Security
  - Disabling HTTP methods not Desired
  - Prevent clickjacking attacks, injection of x-frames
  - X-XSS Protection
  - Implement WAF Mod Security

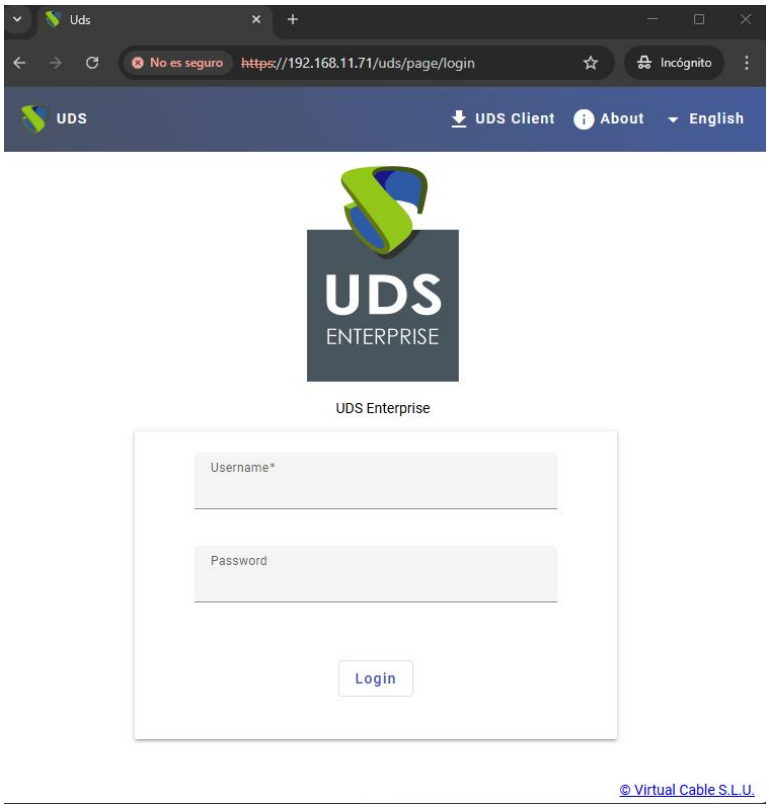
It is advisable to disable SSH access to this server, so that it is only accessible by console.

3.2.4 UDS Actor

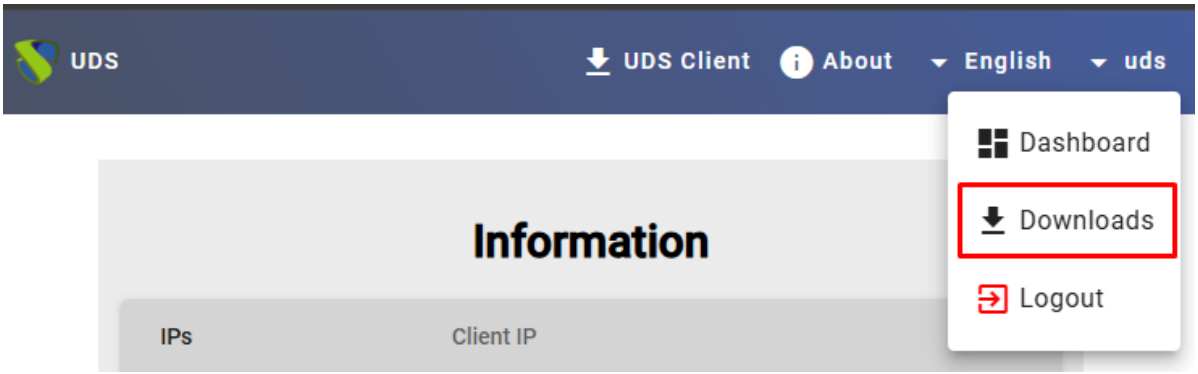
The UDS Actor is a software component that is installed in Windows or Linux OS templates (gold image), which will be used for the deployment of virtual desktops, in RDS application servers to be able to present virtual application sessions to users and in static computers where it is necessary to control user sessions.

To install the UDS Actor, it is necessary to make a previous download from the UDS server (broker) itself, selecting the appropriate Actor for each type of deployment.

To download it, a connection is made to the UDS server through a web browser and with user credentials with **administration permissions** to be able to access the downloads:

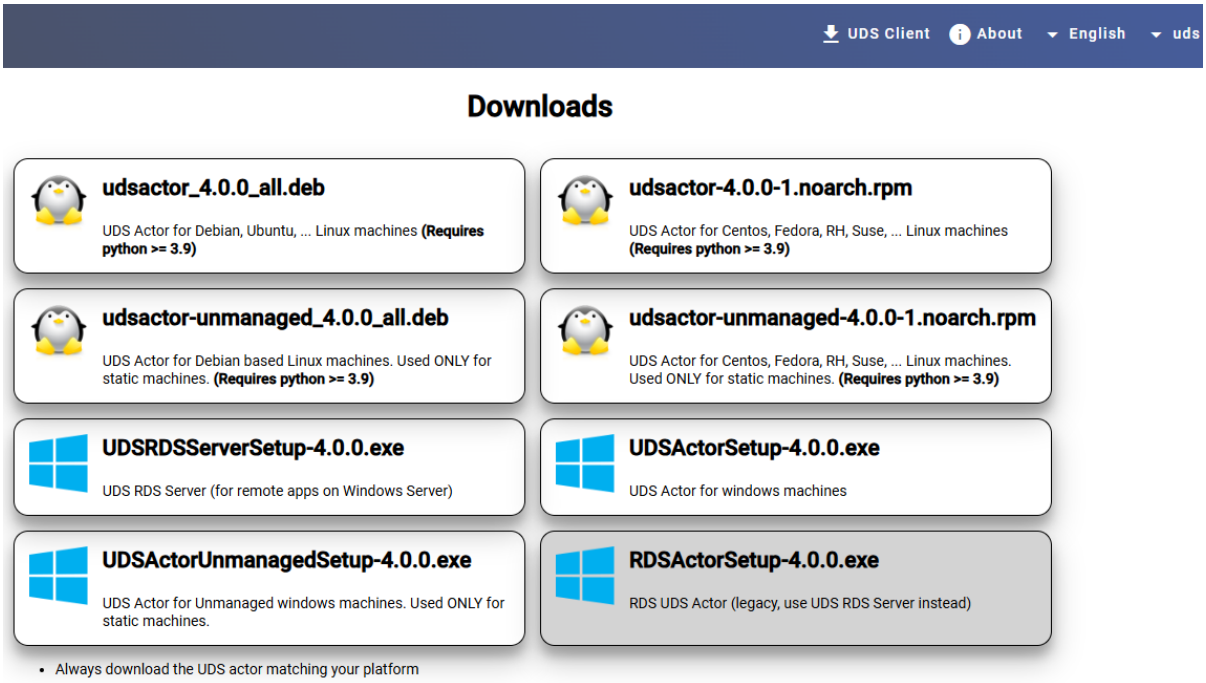


In the user menu we select "Downloads":



The UDS Actors that are available for download will be displayed in the browser. We will select the Actor corresponding to the operating system and service that is installed in the base template or application server on which the desktop services are going to be deployed:

- **udsactor_4.0.0_all.deb:** UDS Actor for Linux Template (Gold Image) machines based on Debian distributions, such as: Ubuntu, Xubuntu, etc…
- **udsactor-4.0.0-1.noarch.rpm:** UDS Actor for Linux Distribution-Based Template (Gold Image) Machines Red Hat or Suse, like: Fedora, OpenSuse, etc…
- **udsactor-unmanaged_4.0.0_all.deb:** UDS actor to control the machine sessions of the service provider "***Static IP Machines Provider***" or the base services of type "***Fixed Machines***". Debian-based distributions, such as: Ubuntu, xUbuntu, etc…
- **udsactor-unmanaged-4.0.0-1.noarch.rpm:** UDS actor to control the machine sessions of the service provider "***Static IP Machines Provider***" or the base services of type "***Fixed Machines***". Distributions based on Red Hat or Suse, like: Fedora, OpenSuse, etc…
- **UDSRDSServerSetup-4.0.0.exe:** Server agent to manage Windows application servers and provide application sessions to users. Windows Server versions 2025, 2019 and 2022.
- **UDSActorSetup-4.0.0.exe:** UDS Actor for Machines template (gold image) Windows. Versions Windows Server and Desktop.
- **UDSActorUnmanagedSetup-4.0.0.exe:** Actor UDS para controlar las sesiones de máquinas Windows del proveedor de servicios "***Static IP Machines Provider***" o los servicios base de tipo "***Fixed Machines***". Versiones Windows Server y Desktop.
- **RDSActorSetup-4.0.0.exe:** UDS (legacy) actor for Windows 2016, 2019, and 2022 application servers with RDS role configured.



Safe Employment Procedure:

Verify that the component being downloaded is the one selected, verifying the certificate if you have it installed.

All actors are their software signed, verify the signature corresponds to the actor.

Enable SSL verification of the server certificate at the time of actor installation.

Enable all the security options that you consider necessary on the machine where the actor is installed.

In addition to the installation of the UDS Actor, it will be necessary to have installed and enabled the connection protocol to be used to connect with the generated desktops (for example, having remote desktop enabled, PCoIP client installed, etc...).

Disable all sharing options via protocol that you do not consider to be strictly necessary for the base machine where you are being installed.

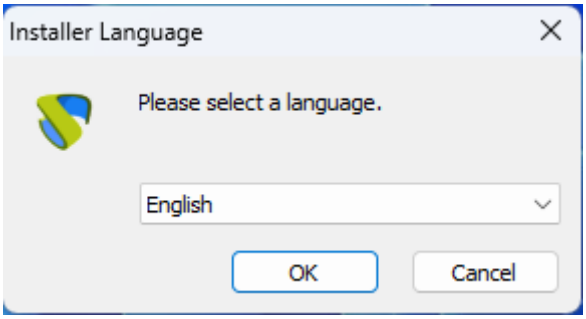
3.2.4.1 Windows Auto-Generated Virtual Desktops

To manage the life cycle of Windows virtual desktops autogenerated by UDS Enterprise, it is necessary that the template machine (gold image) on which they will be based, has the UDS Actor installed: ***UDSActorSetup-4.0.0.exe***

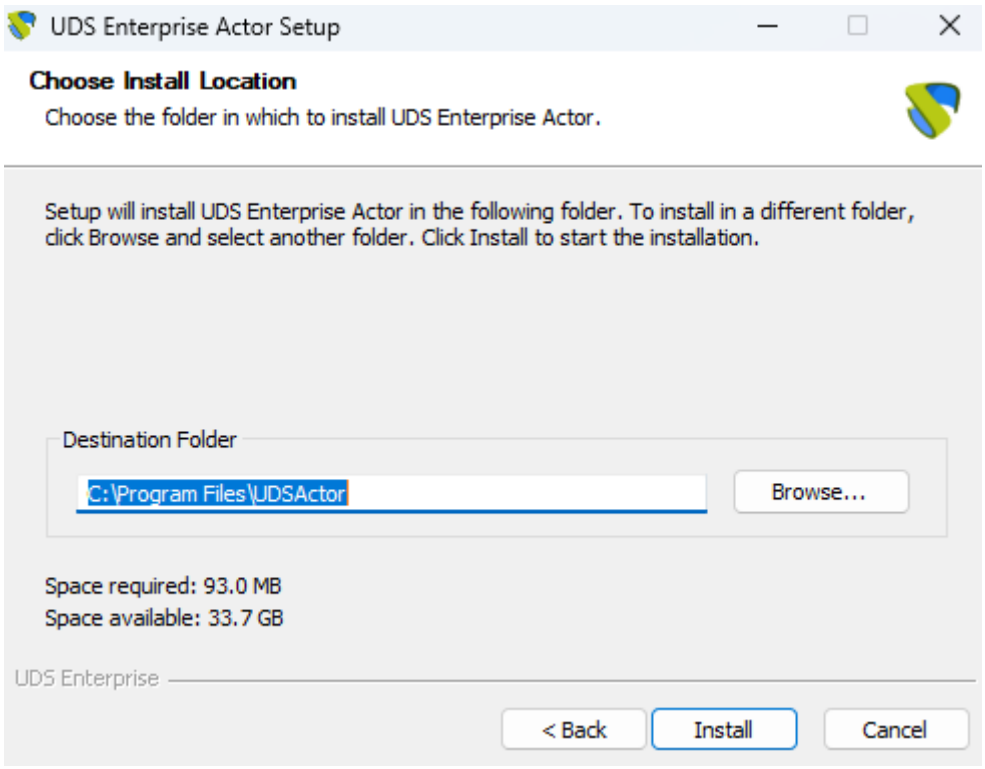
NOTE: Before installing the UDS Actor, it will be necessary to have the IP address or name of the UDS server, the credentials of a user with administration permissions on the UDS environment and at least one authenticator registered in the system.

Once the UDS Actor for Windows OS has been downloaded and transferred to the template machine, we will run it with administration permissions to proceed with its installation.

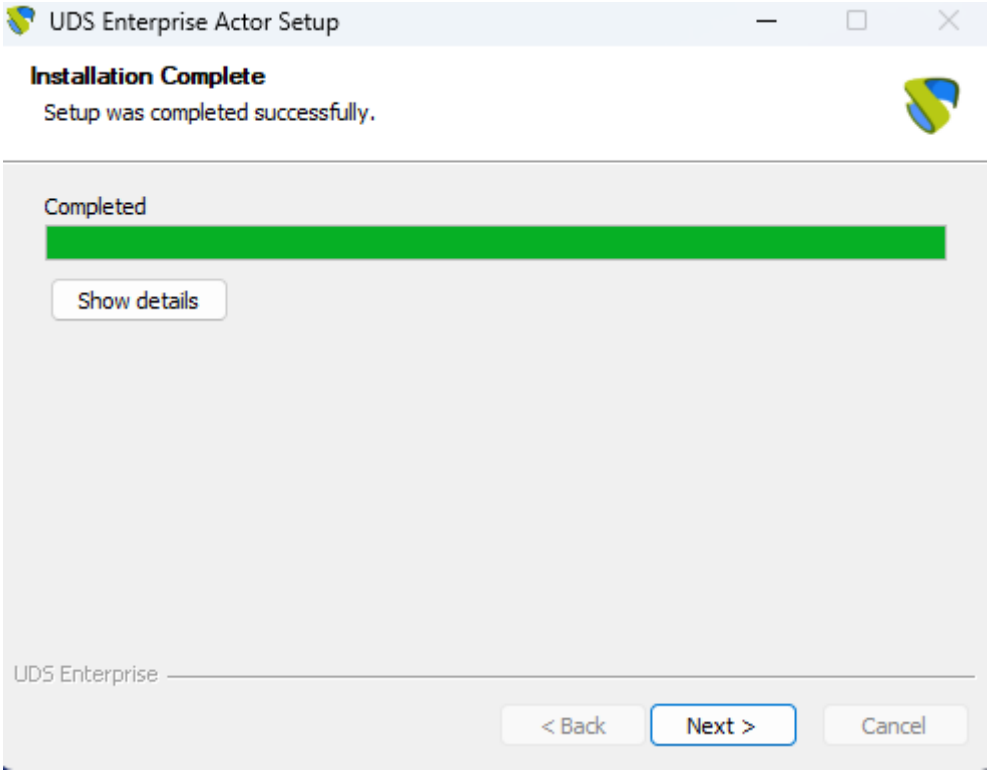
Select the installer language



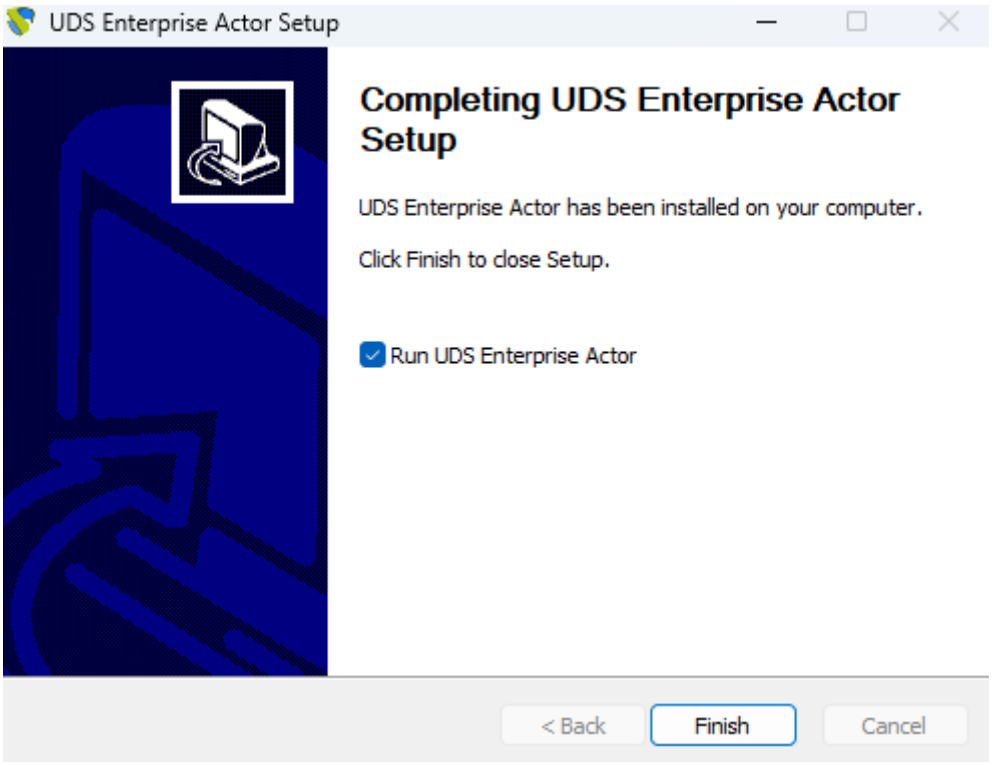
Indicate the installation path of the UDS Actor:



Click on **"Install"** to proceed with the installation:



Once the installation is done, the UDS Actor is configured:



In the ***UDS Server tab*** we will perform the registration of the Actor with the UDS instance indicating the following parameters:

SSL Validation: Type of security applied in communication with the UDS server.

It is recommended to enable the Verify Certificate to obtain the highest possible security".

UDS Server: UDS server name or IP address.

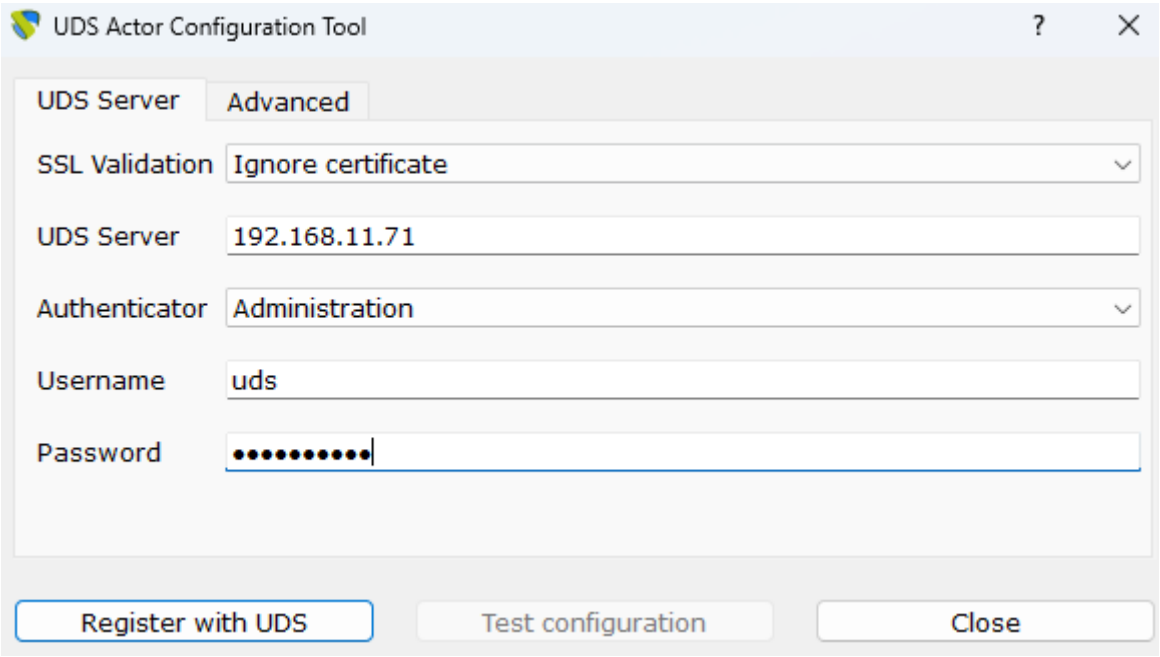
Authenticator: Authenticator to which the administrator user indicated to register the UDS Actor belongs.

It is necessary that the communication with the UDS server is carried out correctly so that the different authenticators are displayed. If no authenticators have been created, it is possible to use the "***Administration***" authenticator that manages the super-user created in the UDS server configuration wizard.

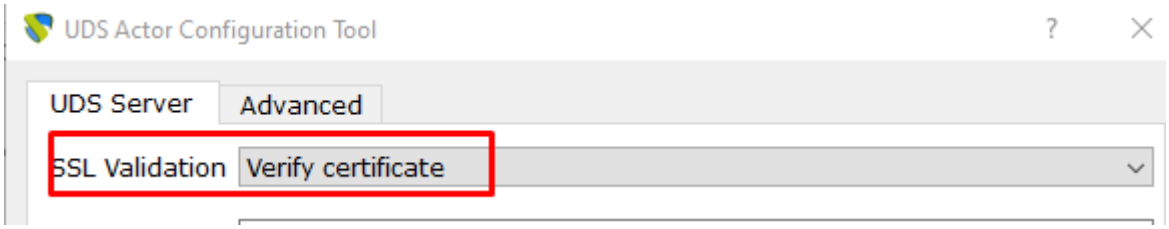
Username: Username with admin permissions in the UDS environment (must belong to the authenticator selected above).

Password: Password of the administrator user used.

Safe use procedure: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.



NOTE: To provide the platform with the greatest possible security, the "Verify Certificate" option must be selected in order to verify that the UDS Server certificate is valid.



In the "***Advanced***" tab we can indicate the following advanced parameters:

Preconnect: script that will be executed just before allowing the user to connect to the virtual desktop.

UDS will automatically pass the following variables, which can be used in the script: username (user indicated in the transport to log in), protocol (rdp, nx, pcoip...), IP (IP that you have recognized in the client (SRC IP)), hostname (SRC Host) username (username that has logged in to the portal).

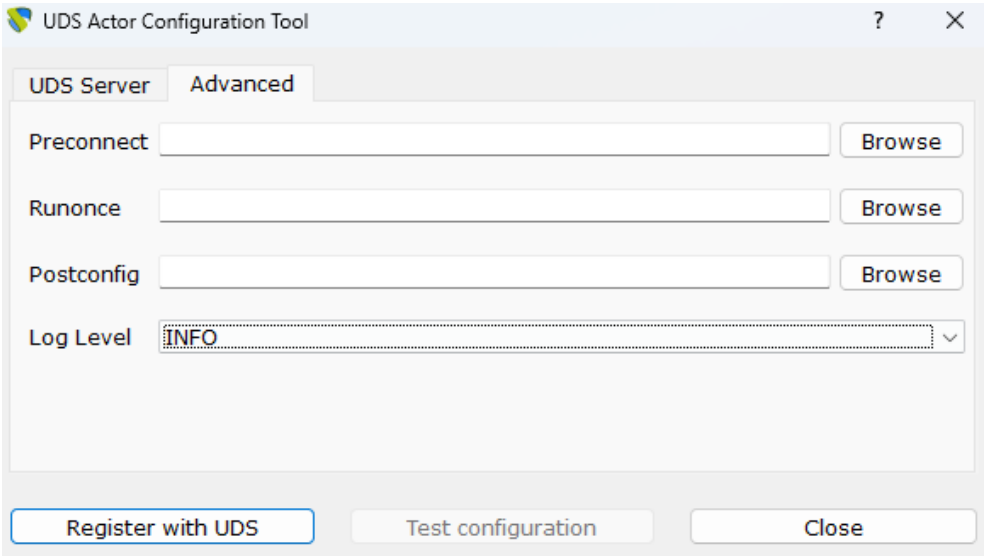
Runonce: script that is executed only once and before the UDS Actor applies its configuration (very useful for example to execute a sysprep to each virtual desktop generated). After its execution it is deleted from the configuration. Parameters can be passed directly to it.

The script that runs must end up restarting the virtual desktop. Otherwise, the desktop will never apply the Actor configuration, preventing reaching the "***Valid***" state in the UDS administration.

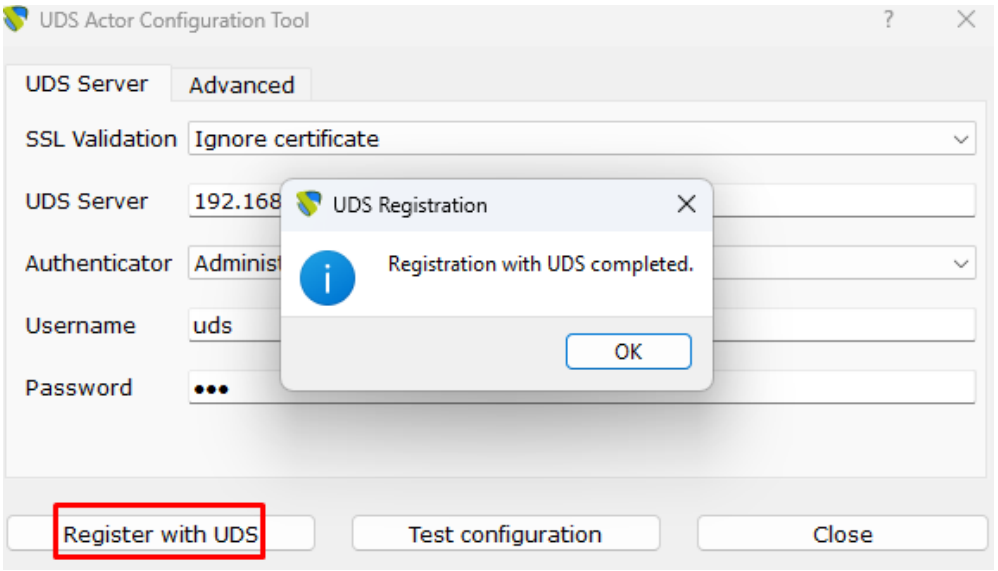
Postconfig: script that runs when the UDS Actor has finished its configuration. Parameters can be passed directly to it.

The script runs only once, but unlike "Runonce" mode you don't need to restart the virtual desktop. This script is useful to add some "own" element to the configuration made by the UDS Actor, such as copying files from the local network, executing configurations, etc…

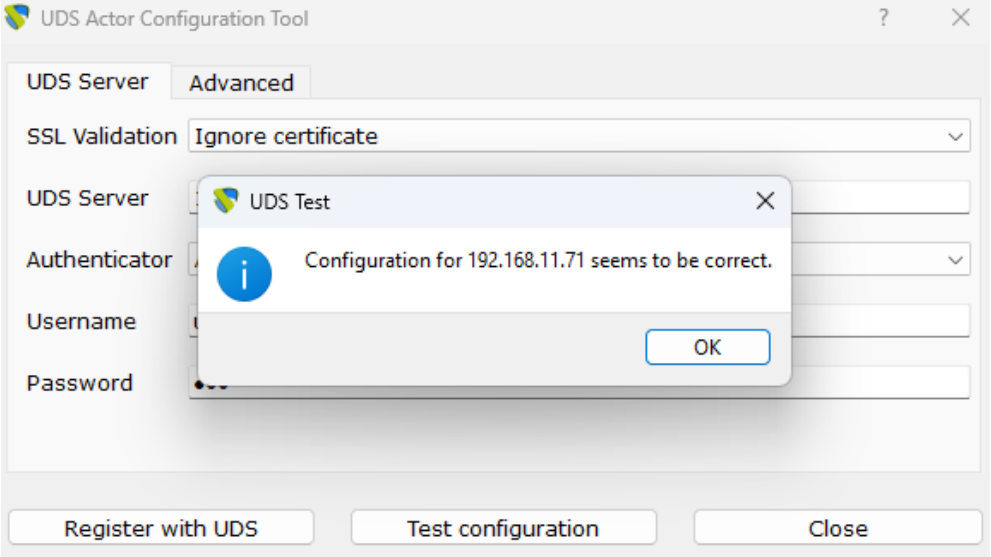
Log Level: Types of logs that will be displayed in the UDS Actor log files. These log files (udsactor.log) will be located in the paths: %temp% (path of the user's temporary files) and C:\Windows\Temp (path of the temporary files of the OS).).



Once all the necessary data has been entered, we will click on "Register with UDS":



We can also perform a test by clicking on "Test configuration" to verify the correct connection with the UDS server at any time:



It is very important to bear in mind that if any data is modified it will always be necessary to carry out the registration process afterwards (by clicking on the "Register with UDS" button), if this action is not carried out, the changes will not be applied.

Once the installation and configuration of the UDS Actor is done, the template machine (gold image) can be turned off and will be available to be used by UDS to auto-generate virtual desktops.

NOTE: In addition to the installation of the UDS Actor, it will be necessary to have enabled the connection protocol to be used to connect with the generated desktops (for example, to have the remote desktop enabled, the PCoIP agent, NoMachine, etc.…).

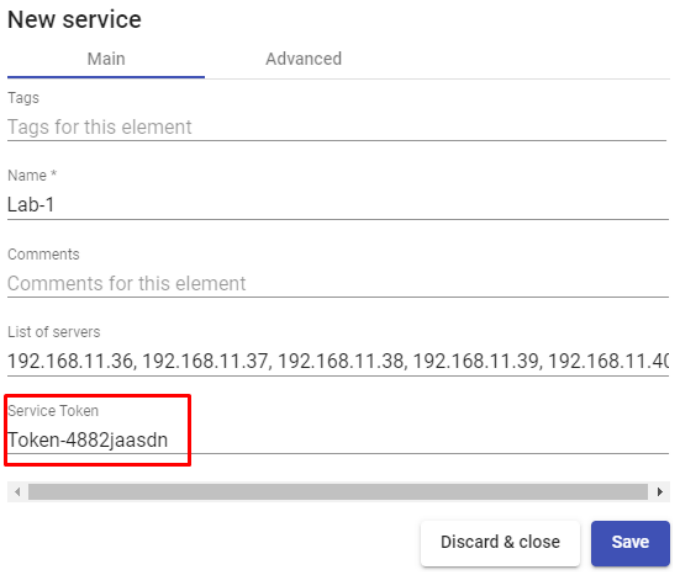
3.2.4.2 Windows Static Desktops

To control the user sessions (login and logout) of an existing machine configured within the "***Static IP Machines Provider***" or the base services of type "***Fixed Machines***", it is necessary that it has the UDS Actor installed: ***UDSActorUnmanagedSetup-4.0.0.exe***

In addition to controlling a user's session, the actor will detect if there is already a user connected (for example, when we access a physical computer) and will prevent the connection of another who has requested access.

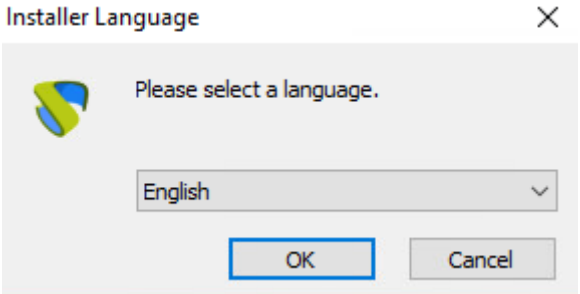
If these machines do not have the Actor installed and are part of a "***Static Multiple IP***" type service or the "***Fixed Machines" base services***, UDS will not be able to control the user's logout of the machine and, therefore, will not be able to free it to make it available to another user.

NOTE: Before installing the UDS Actor, it will be necessary to have the IP address or name of the UDS server and the "Service Token" key registered in a "Static Multiple IP" service within the "Static IP Machines Provider" service provider or the base services of the "Fixed Machines" type"

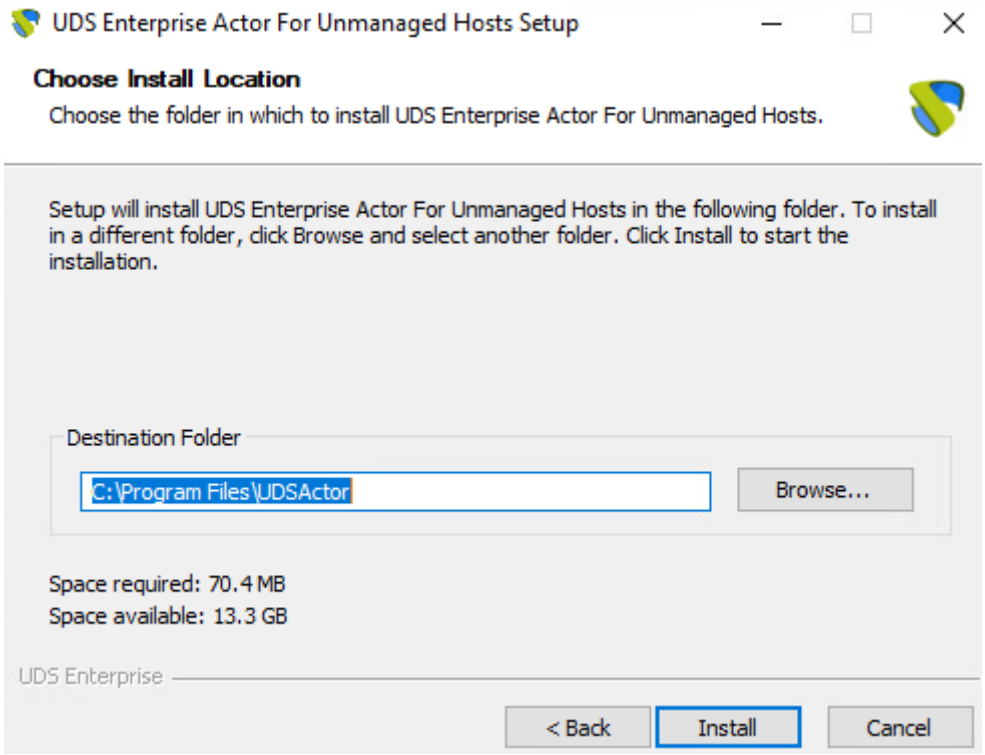


Once the UDS Actor for Windows OS has been downloaded and transferred to the machine to which we want to connect users (whether physical or virtual), we will run it to proceed with its installation.

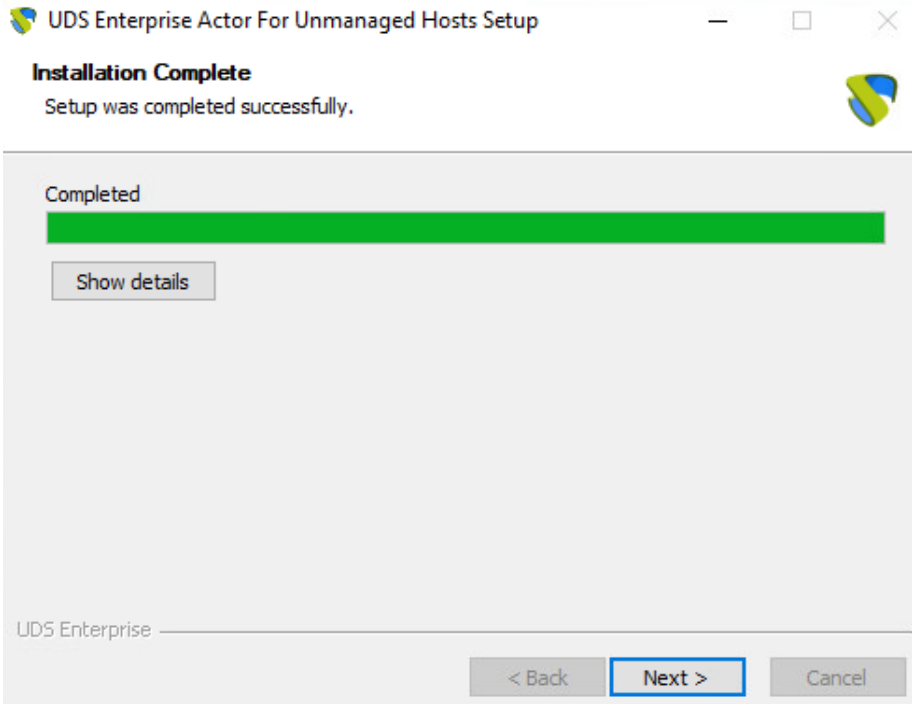
Select the installer language:



Indicate the installation path of the UDS Actor:



Click on **"Install"** to proceed with the installation:



Once the installation is done, the UDS Actor is configured:

We proceed to register the Actor with the UDS server indicating the following parameters:

SSL Validation: Type of security applied in communication with the UDS server.

It is recommended to obtain the highest possible security, activate the verification of the certificate "**Verify Certificate**".

UDS Server: UDS server name or IP address.

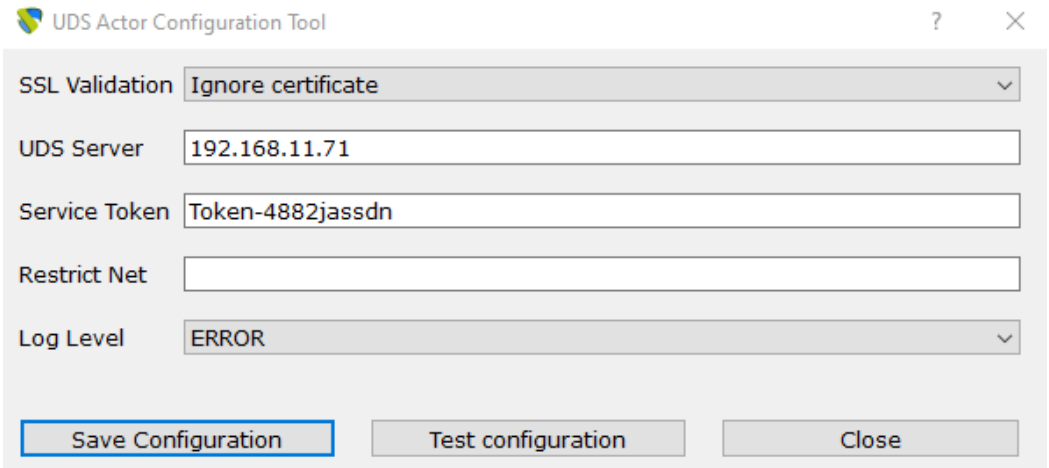
Service Token: Code created in UDS administration, in the "***Static Multiple IP***" service type within the "***Static IP Machines Provider***" service provider or the base services of type "***Fixed Machines***".

Log Level: Types of logs that will be displayed in the UDS Actor log files. These log files (udsactor.log) will be located in the paths: %temp% (path of the user's temporary files) and C:\Windows\Temp (path of the temporary files of the OS.).

Restrict Net: Adds the possibility of discriminating networks for connection with UDS.

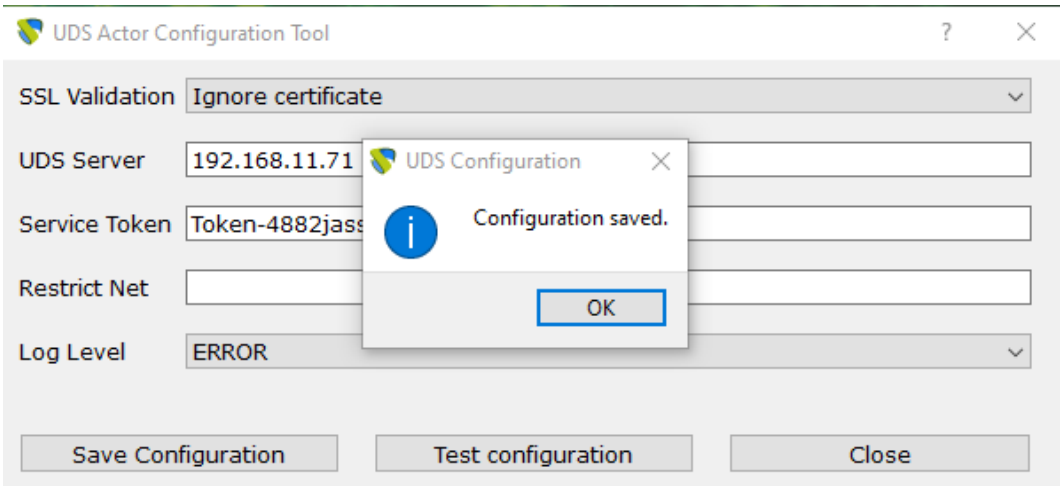
The network that we indicate will be used to notify the user of access to the machine.

Necessary for cases where we have more than one network card, **otherwise leave blank**.

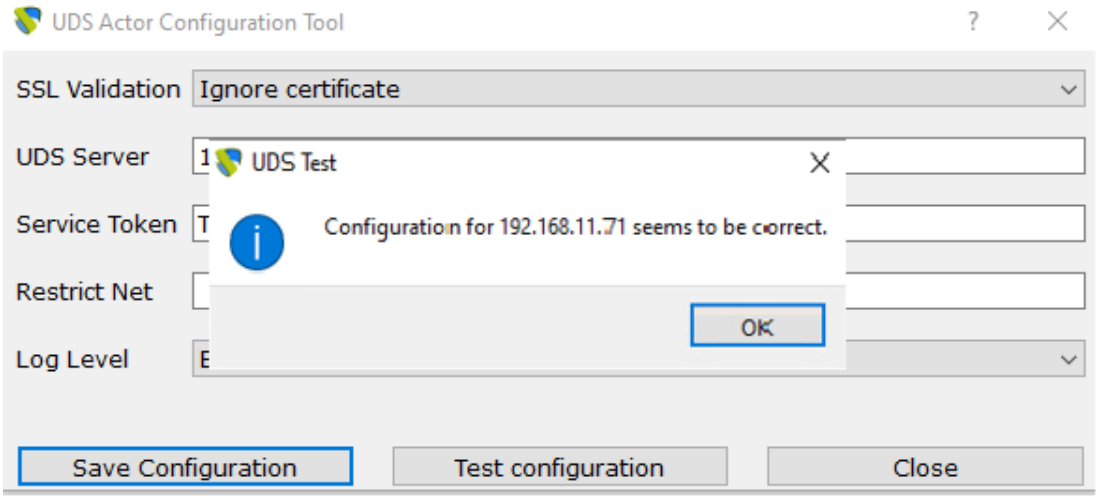


Note: To provide the platform with the greatest possible security, the "Verify Certificate" option must be selected to verify that the UDS Server certificate is valid.

Once you have entered this data, click on "***Save Configuration***":



It will be necessary to run the configuration test to check if the indicated data is correct and there is connectivity with the UDS server:



Once the installation and configuration of the UDS Actor has been carried out, **we must restart the machine,** and it will be available to be assigned by UDS and control the user sessions.

NOTE:
In addition to the installation of the UDS Actor, it will be necessary to have enabled the connection protocol to be used to connect with the generated desktops (for example, to have the remote desktop enabled, etc...).

3.2.4.3 Auto-generated Linux virtual desktops

To manage the life cycle of the Linux virtual desktops autogenerated by UDS Enterprise, it is necessary that the template machine (gold image) on which they will be based, has installed the UDS Actor for the different Linux distributions:

- **Debian-based distributions:** ***udsactor_4.0.0_all.deb***
- **Red Hat and Suse-based distributions:** ***udsactor-4.0.0-1.noarch.rpm***

NOTE: Before installing the UDS Actor, it will be necessary to have the IP address or name of the UDS server, the credentials of a user with administration permissions on the UDS environment.

Once the UDS Actor for the chosen Linux distribution has been downloaded and transferred to the template machine, we will run it with administration permissions to proceed with its installation.

It is strongly recommended to perform such execution of the Actor via command console:

```
user@ubuntu24:~/Descargas$ ls
udsactor_4.0.0_all.deb
user@ubuntu24:~/Descargas$ sudo dpkg -i udsactor_4.0.0_all.deb
```

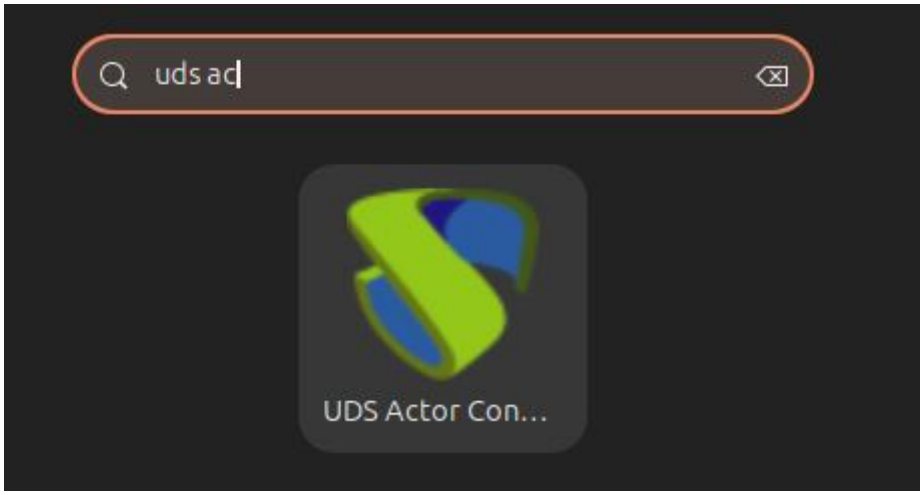
```
user@ubuntu24:~/Descargas$ sudo dpkg -i udsactor_4.0.0_all.deb
[sudo] contraseña para user:
Seleccionando el paquete udsactor previamente no seleccionado.
(Leyendo la base de datos ... 153308 ficheros o directorios instalados actualm
ente.)
Preparando para desempaquetar udsactor_4.0.0_all.deb ...
Desempaquetando udsactor (4.0.0) ...
dpkg: problemas de dependencias impiden la configuración de udsactor:
 udsactor depende de xscreensaver; sin embargo:
  El paquete `xscreensaver' no está instalado.

dpkg: error al procesar el paquete udsactor (--install):
 problemas de dependencias - se deja sin configurar
Procesando disparadores para gnome-menus (3.36.0-1.1ubuntu3) ...
Procesando disparadores para desktop-file-utils (0.27-2build1) ...
Se encontraron errores al procesar:
 udsactor
user@ubuntu24:~/Descargas$
```

If we get an error due to lack of dependencies, we will proceed with its installation:

```
user@ubuntu24:~/Descargas$ sudo apt install -f
```

```
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 25 no actualizados.
1 no instalados del todo o eliminados.
Se necesita descargar 9.901 kB de archivos.
Se utilizarán 26,6 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

Once the necessary dependencies have been installed, the installation of the UDS actor will also be carried out automatically. Now we will run the UDS Actor configuration:



In the ***UDS Server tab*** we will perform the registration of the Actor with the UDS instance indicating the following parameters:

SSL Validation: Type of security applied in communication with the UDS server.

It is recommended to obtain the highest possible security, activate the verification of the certificate "**Verify Certificate**".

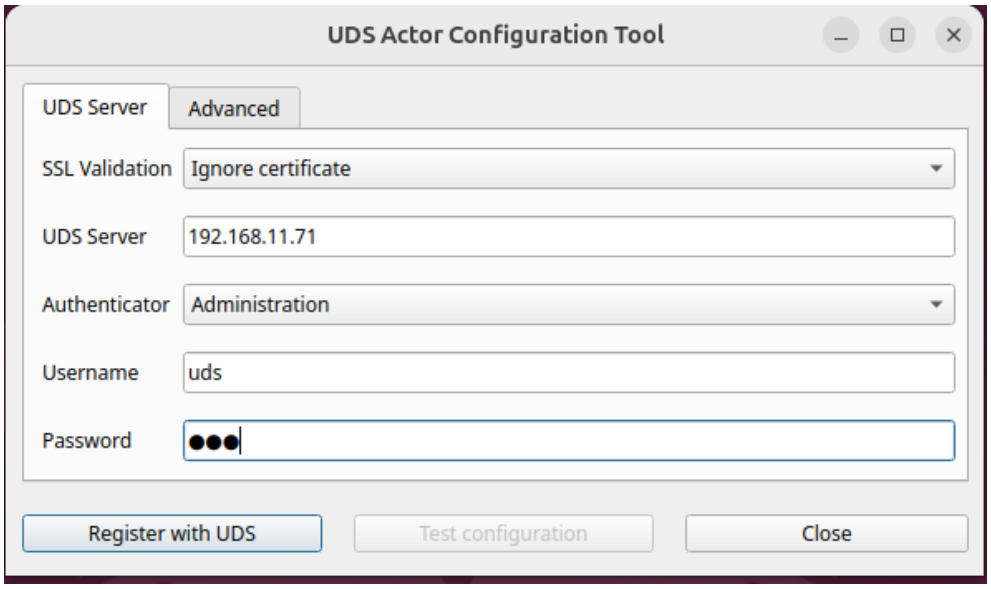
UDS Server: UDS server name or IP address.

Authenticator: Authenticator to which the administrator user indicated to register the UDS actor belongs.

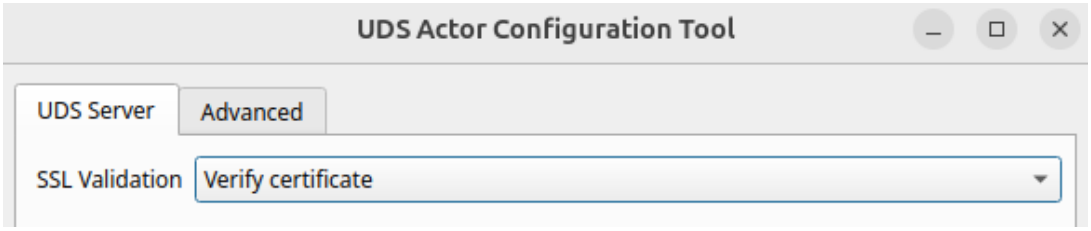
It is necessary that the communication with the UDS server is carried out correctly so that the different authenticators are displayed. If no authenticators have been created, it is possible to use the "***Administration***" authenticator that manages the super-user created in the UDS server configuration wizard.

Username: Username with admin permissions in the UDS environment (must belong to the authenticator selected above).

Password: Password of the administrator user used.



Note: To provide the platform with the greatest possible security, the "Verify Certificate" option must be selected to verify the certificate.



In the "***Advanced***" tab, we can indicate the following advanced parameters:

Preconnect: Script that will be executed just before allowing the user to connect to the virtual desktop.

UDS will automatically pass the following variables, which can be used in the script: username (user indicated in the transport to log in), protocol (rdp, nx, pcoip...), IP (IP that you have recognized in the client (SRC IP)), hostname (SRC Host) username (username that has logged in to the portal).

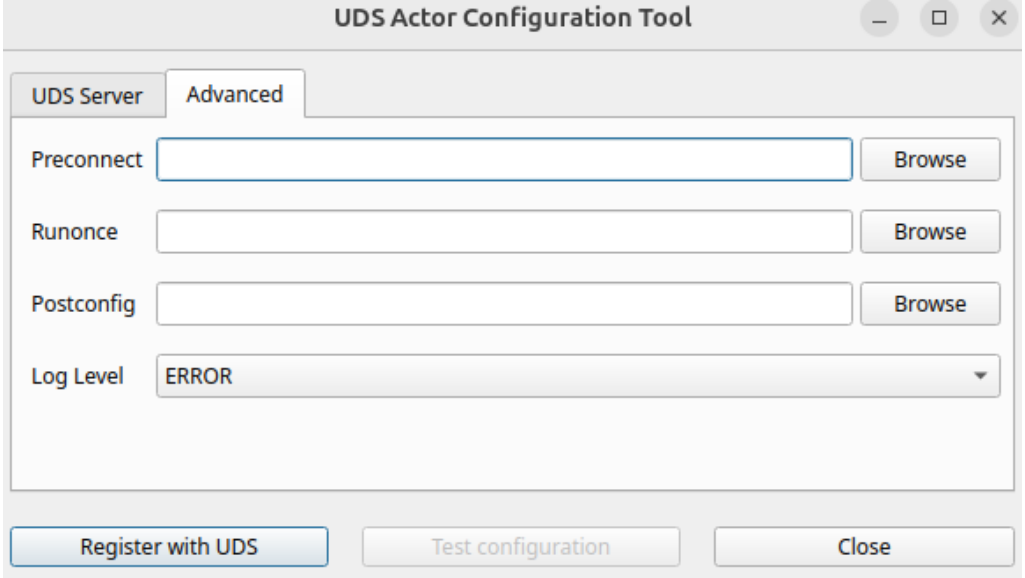
Runonce: A script that is executed only once and before the UDS Actor applies its settings. After its execution it is deleted from the configuration. Parameters can be passed directly to it.

The script that runs must end up restarting the virtual desktop. Otherwise, the desktop will never apply the Actor configuration preventing it from reaching its "***Valid***" state in the UDS administration.

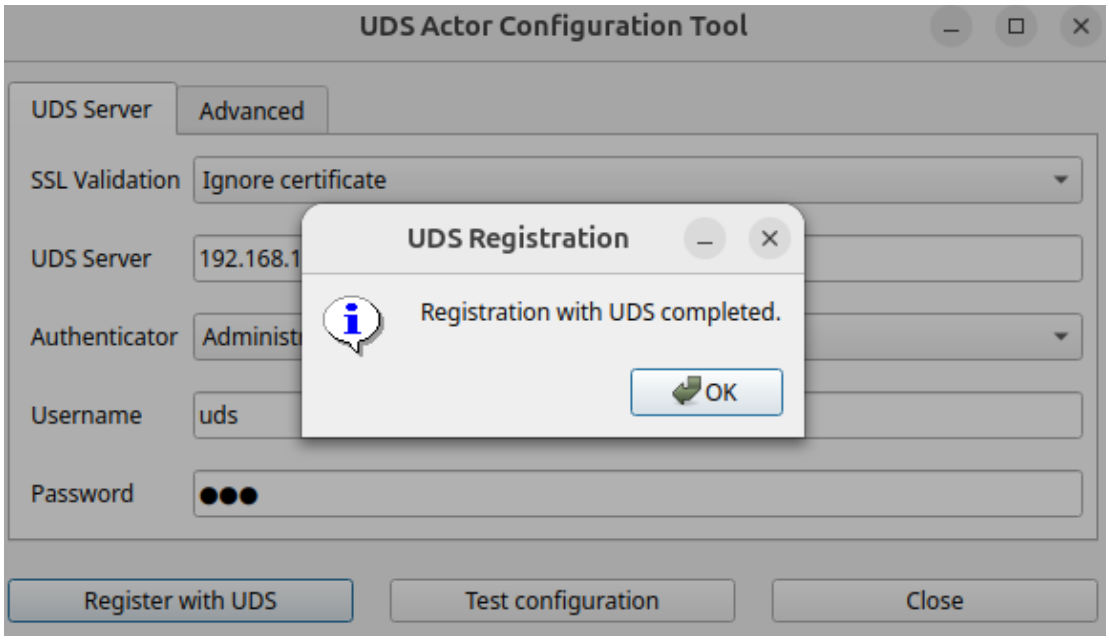
Postconfig: A script that is executed when the UDS Actor has finished its configuration. Parameters can be passed directly to it.

The script is executed only once, but unlike "Runonce" mode it does not need to restart the virtual desktop. This script is useful to add some "own" element to the configuration made by the UDS Actor, such as copying files from the local network, executing configurations, etc…

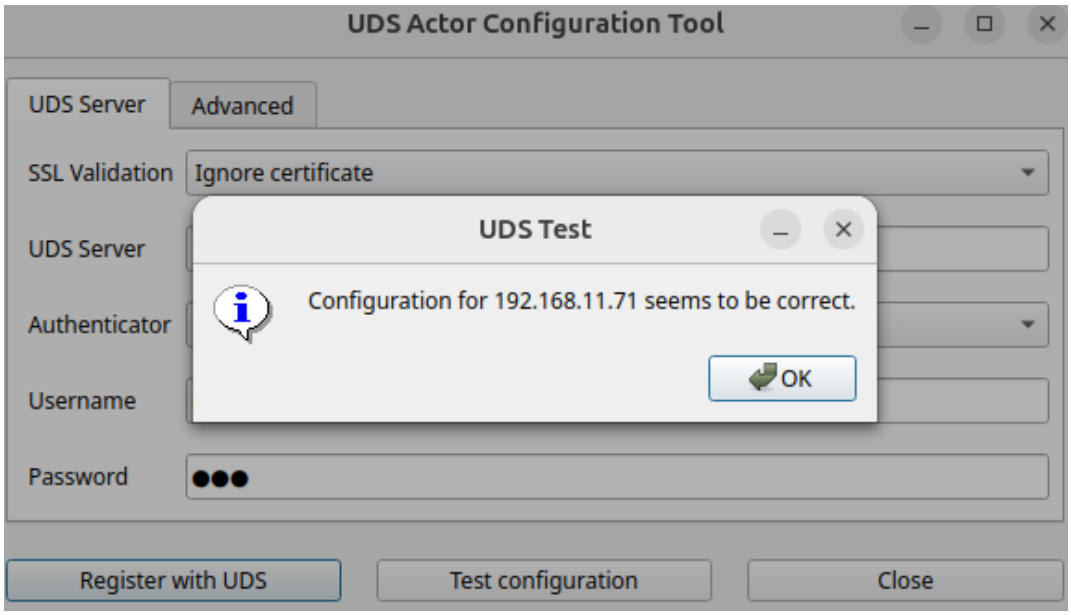
Log Level: Types of logs that will be displayed in the UDS Actor log files. These log files (udsactor.log) will be located in the path: /var/log/



Once you have entered this data, click on "Register with UDS":



We can also perform a test by clicking on "Test configuration" to verify the correct connection with the UDS server at any time:



It is very important to bear in mind that if any data is modified it will always be necessary to carry out the registration process afterwards (by clicking on the "Register with UDS" button), if this action is not carried out, the changes will not be applied.

Once the installation and configuration of the UDS Actor is done, the template machine (gold image) can be turned off and will be available to be used by UDS to auto-generate virtual desktops.

NOTE: In addition to the installation of the UDS Actor, it will be necessary to have enabled the connection protocol to be used to connect with the generated desktops (for example, to have XRDP, X2Go Server, NoMachine etc installed and enabled...).

It will also be necessary to make sure that the screensaver is installed, but not enabled, in addition to **not having "auto login" enabled** on the machine.

3.2.4.4 Linux Static Desktops

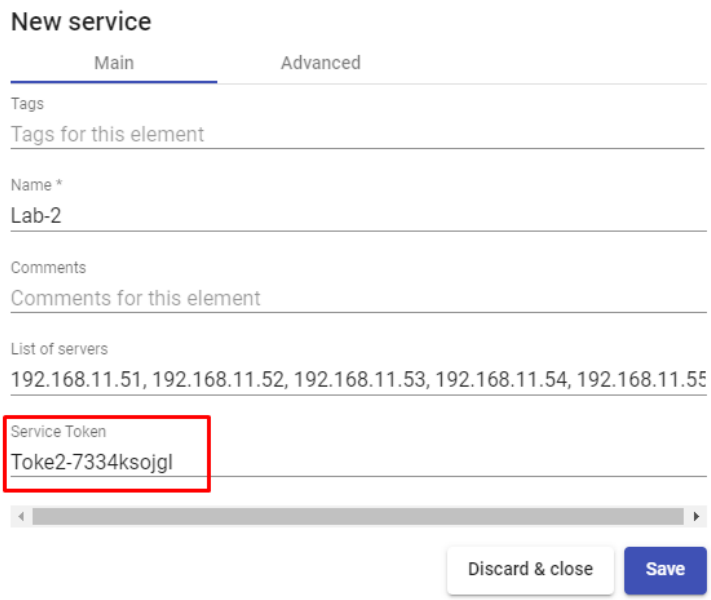
To control the user sessions (login and logout) of an existing machine configured within the "***Static IP Machines Provider***" or the base services of type "***Fixed Machines***", it is necessary that it has the UDS Actor installed for the different Linux distributions:

- **Debian-based distributions:** ***udsactor-unmanaged_4.0.0_all.deb***
- **Red Hat and Suse-based distributions:** ***udsactor-unmanaged-4.0.0-1.noarch.rpm***

In addition to controlling a user's session, the actor will detect if there is already a user connected (for example, when we access a physical computer) and will prevent the connection of another who has requested access.

If these machines do not have the UDS Actor installed and are part of a "***Static Multiple IP***" service or the base services of the "***Fixed Machines***" type, UDS will not be able to control the user's logout of the machine and, therefore, will not be able to release it to make it available to another user.

NOTE: Before installing the UDS Actor, it will be necessary to have the IP address or name of the UDS server and the "Service Token" key registered in a "Static Multiple IP" service within the "Static IP Machines Provider" service provider"



Once the UDS Actor for Linux OS has been downloaded and transferred to the machine to which we want to connect users (whether physical or virtual), we will run it to proceed with its installation.

It is strongly recommended to perform such execution of the Actor via command console:

```
user@ubuntu24:~/Descargas$ sudo dpkg -i udsactor-unmanaged_4.0.0_all.deb
[sudo] contraseña para user:
Seleccionando el paquete udsactor-unmanaged previamente no seleccionado.
```

If we get an error due to lack of dependencies, we will proceed with its installation:

```
Preparando para desempaquetar udsactor-unmanaged_4.0.0_all.deb ...
Desempaquetando udsactor-unmanaged (4.0.0) ...
dpkg: problemas de dependencias impiden la configuración de udsactor-unmanaged:
 udsactor-unmanaged depende de xscreensaver; sin embargo:
  El paquete `xscreensaver' no está instalado.

dpkg: error al procesar el paquete udsactor-unmanaged (--install):
 problemas de dependencias - se deja sin configurar
Procesando disparadores para gnome-menus (3.36.0-1.1ubuntu3) ...
Procesando disparadores para desktop-file-utils (0.27-2build1) ...
Se encontraron errores al procesar:
 udsactor-unmanaged
user@ubuntu24:~/Descargas$ sudo apt install -f
```

```
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 25 no actualizados.
1 no instalados del todo o eliminados.
Se necesita descargar 9.901 kB de archivos.
Se utilizarán 26,6 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

Once the necessary dependencies have been installed, the installation of the UDS actor will also be carried out automatically. Now we will run the UDS Actor configuration:

Proceed to register the Actor with the UDS server indicating the following parameters:

SSL Validation: Type of security applied in communication with the UDS server.

It is recommended to obtain the highest possible security, activate the verification of the certificate "**Verify Certificate**".

UDS Server: UDS server name or IP address.

Service Token: Code created in UDS administration, in the "***Static Multiple IP***" service type within the "***Static IP Machines Provider***" service provider or the base services of type "***Fixed Machines'***".

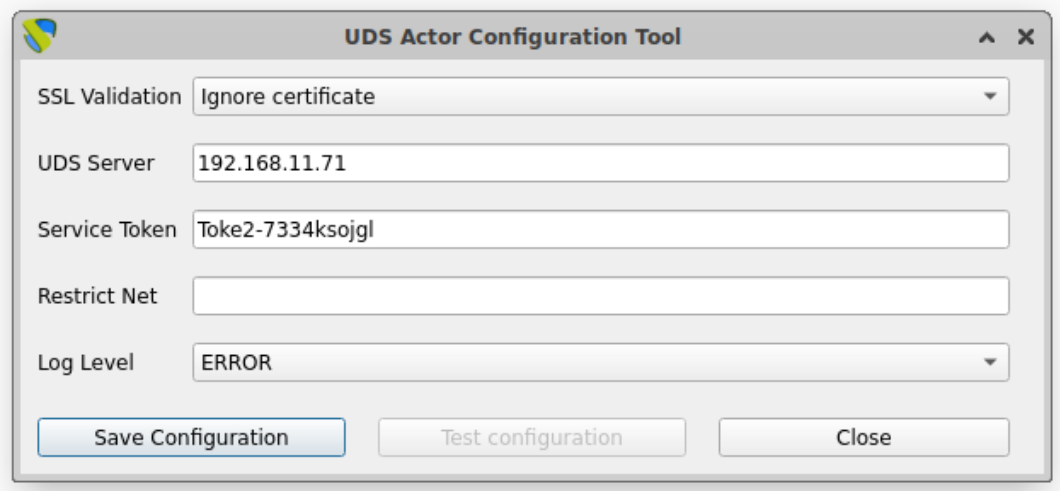
Log Level: Types of logs that will be displayed in the UDS Actor log files. These log files (udsactor.log) will be located in the path: /var/log/

Restrict Net: Adds the possibility of discriminating networks for connection with UDS.

The network that we indicate will be used to notify the user of access to the machine.

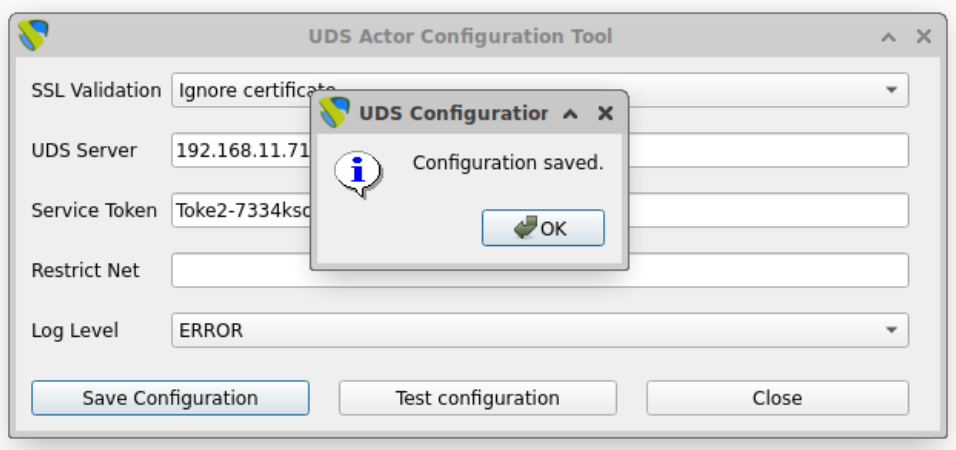
Necessary for cases where we have more than one network card, otherwise leave blankThe network that we indicate will be used to notify the user of access to the machine.

Necessary for cases where we have more than one network card, otherwise leave blank.

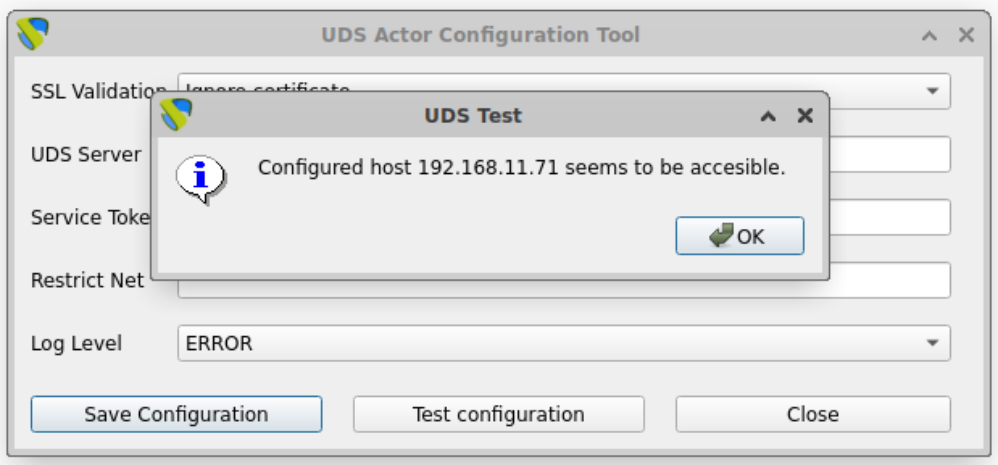


Note: To provide the platform with the greatest possible security, the "Verify Certificate" option must be selected to verify the certificate.

Once you have entered this data, click on "***Save Configuration***":



It will be necessary to run the configuration test to check if the indicated data is correct and there is connectivity with the UDS server:



Once the installation and configuration of the UDS Actor has been carried out, we must restart the machine and it will be available to be assigned by UDS and control the user sessions.

NOTE:
In addition to the installation of the UDS Actor, it will be necessary to have the connection protocol to be used to connect to the generated desktops enabled (For example, to have XRDP installed and enabled).

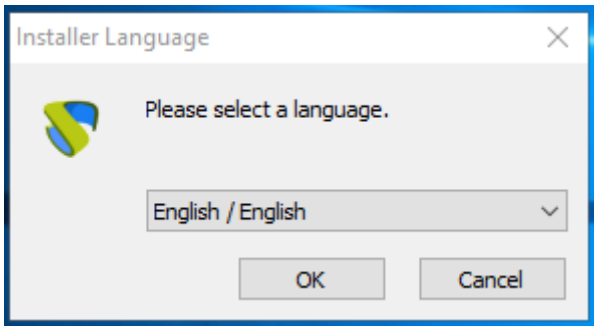
3.2.4.5 Windows Virtual Apps

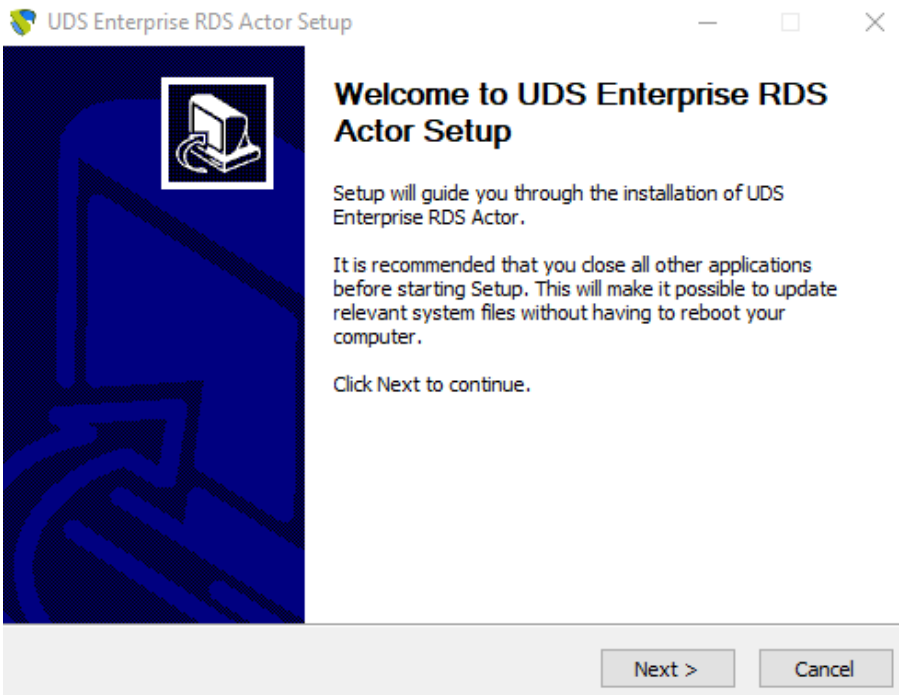
In order for UDS to publish Windows application sessions and control their lifecycle, it is necessary for Windows application servers to have the UDS Actor installed: ***RDSActorSetup-4.0.0.exe***

NOTE:
Before installing the UDS Actor, it will be necessary to have the IP address or name of the UDS server, the credentials of a user with administration permissions on the UDS environment.

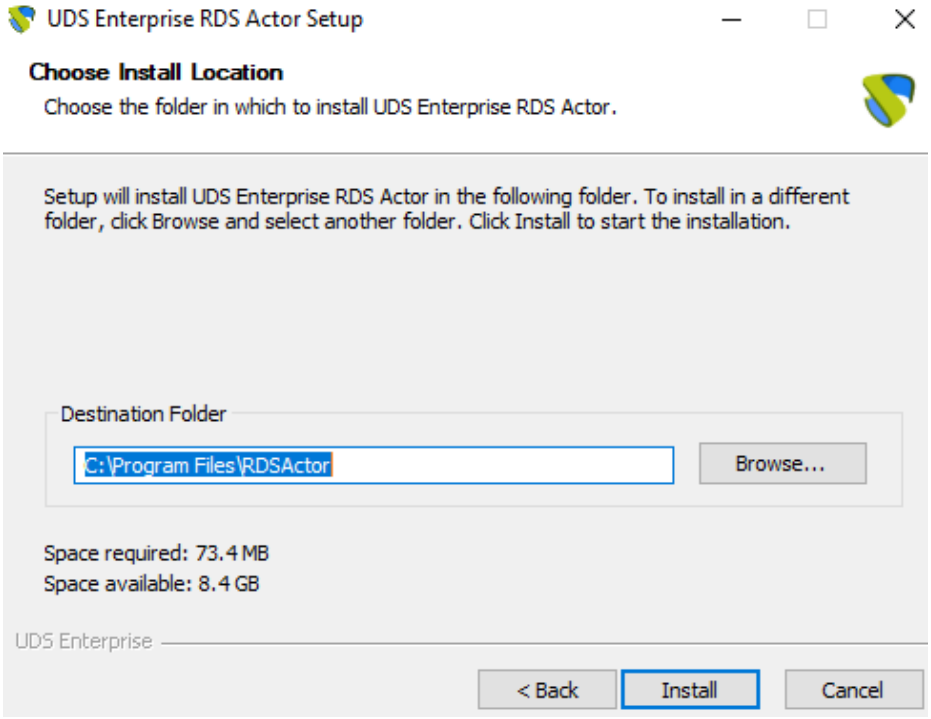
Once the UDS Actor for Windows Server application servers has been downloaded, we run it with administration permissions to proceed with its installation.

Select the installer language:

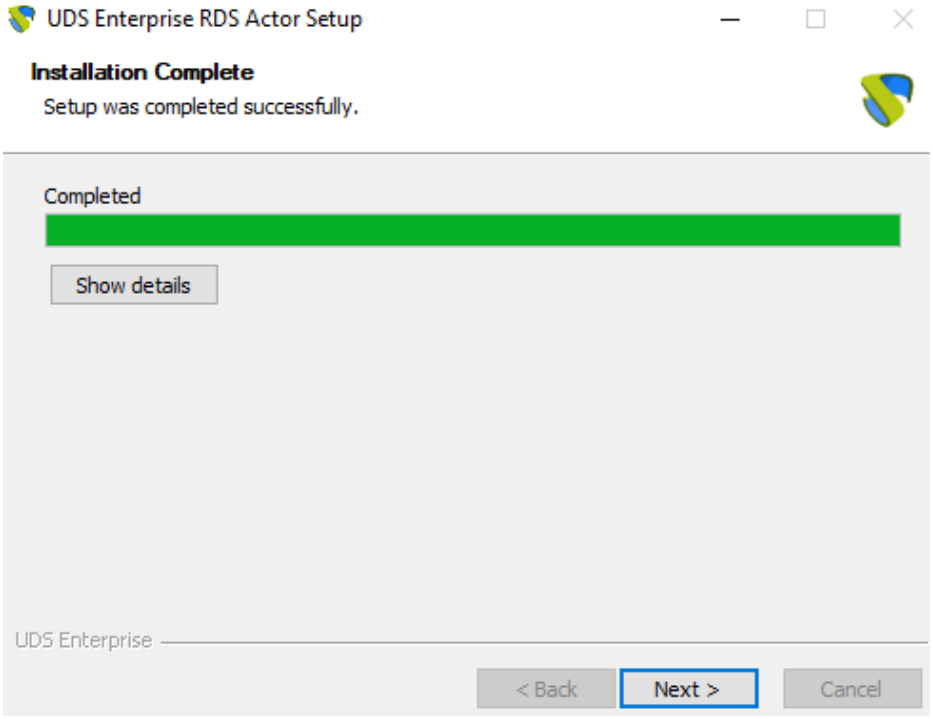




Indicate the installation path of the UDS Actor:



Click on **"Install"** to proceed with the installation:



Once the installation is done, the UDS Actor is configured:

We proceed to register the Actor with the UDS server indicating the following parameters:

SSL Validation: Type of security applied in communication with the UDS server.

It is recommended to obtain the greatest possible security, activate the verification of the certificate "**Verify Certificate**" if this is not the case, there will be a great risk to the security of the platform.

UDS Server: UDS server name or IP address.

Authenticator: Authenticator to which the administrator user indicated to register the UDS Actor belongs.

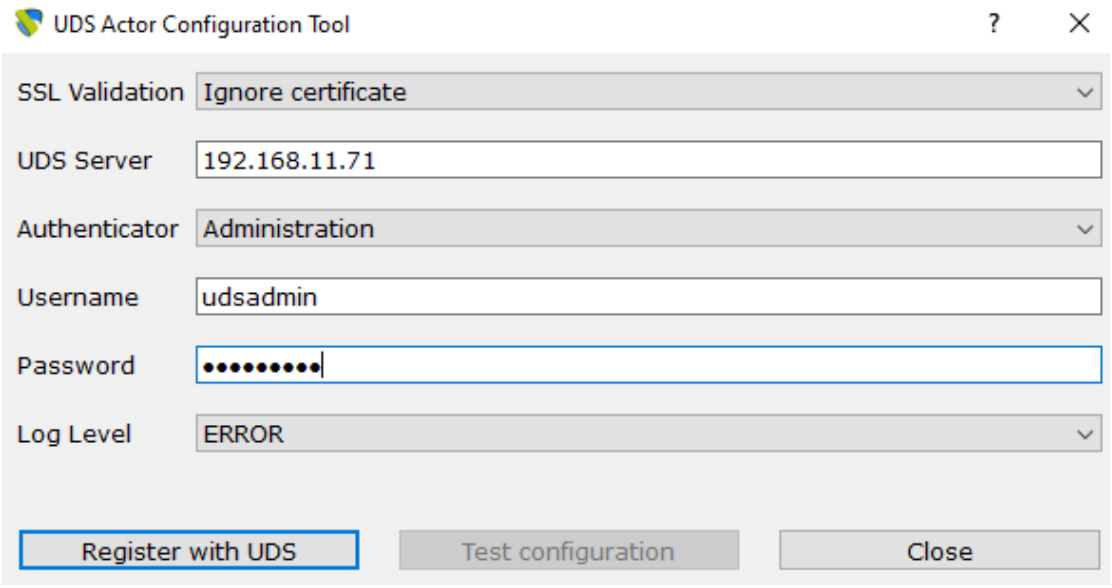
It is necessary that the communication with the UDS server is carried out correctly so that the different authenticators are displayed. If no authenticators have been created, it is possible to use the "***Administration***" authenticator that manages the super-user created in the UDS server configuration wizard.

Username: Username with admin permissions in the UDS environment (must belong to the authenticator selected above).

Password: Password of the administrator user used.

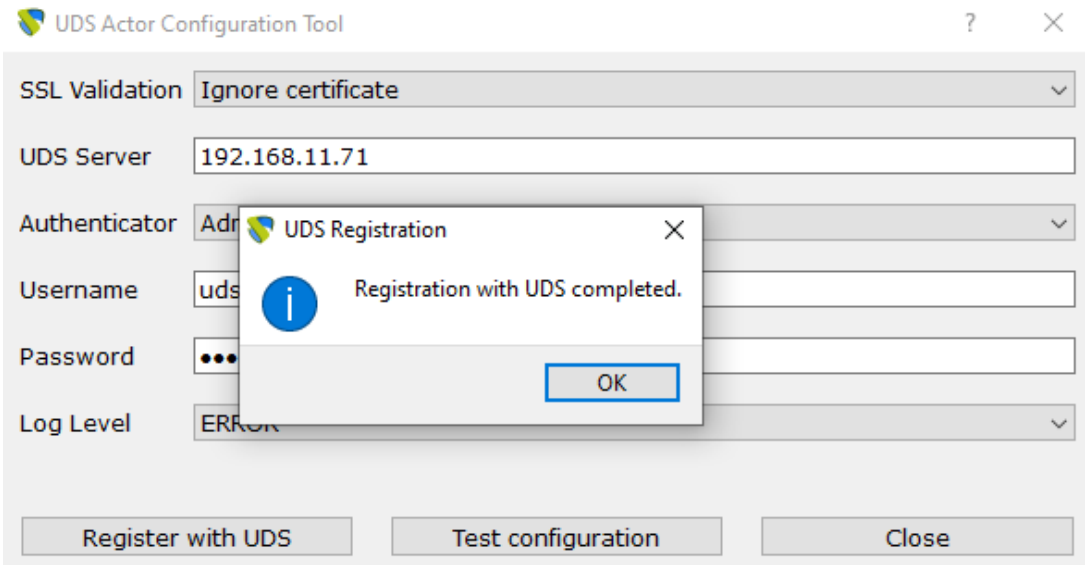
Procedimiento de empleo seguro: Passwords must be of sufficient length and include upper and lower case, numbers and special characters.

Log Level: Types of logs that will be displayed in the UDS Actor log files.

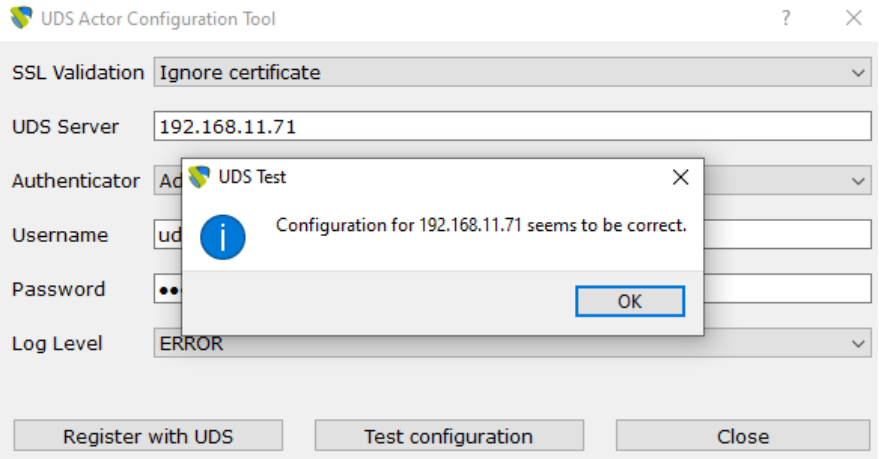


Note: To provide the platform with the greatest possible security, the "Verify Certificate" option must be selected to verify the certificate.

Once all the necessary data has been entered, click on "Register with UDS":



We can also perform a test by clicking on "Test configuration" to verify the correct connection with the UDS server at any time:



It is very important to bear in mind that if any data is modified it will always be necessary to carry out the registration process afterwards (by clicking on the "Register with UDS" button), if this action is not carried out, the changes will not be applied.

Once the UDS Actor for RDS servers has been installed and configured, we will be able to install the Remote Desktop Session (RDS) feature and integrate it with the UDS Actor in order to provide application sessions to users.

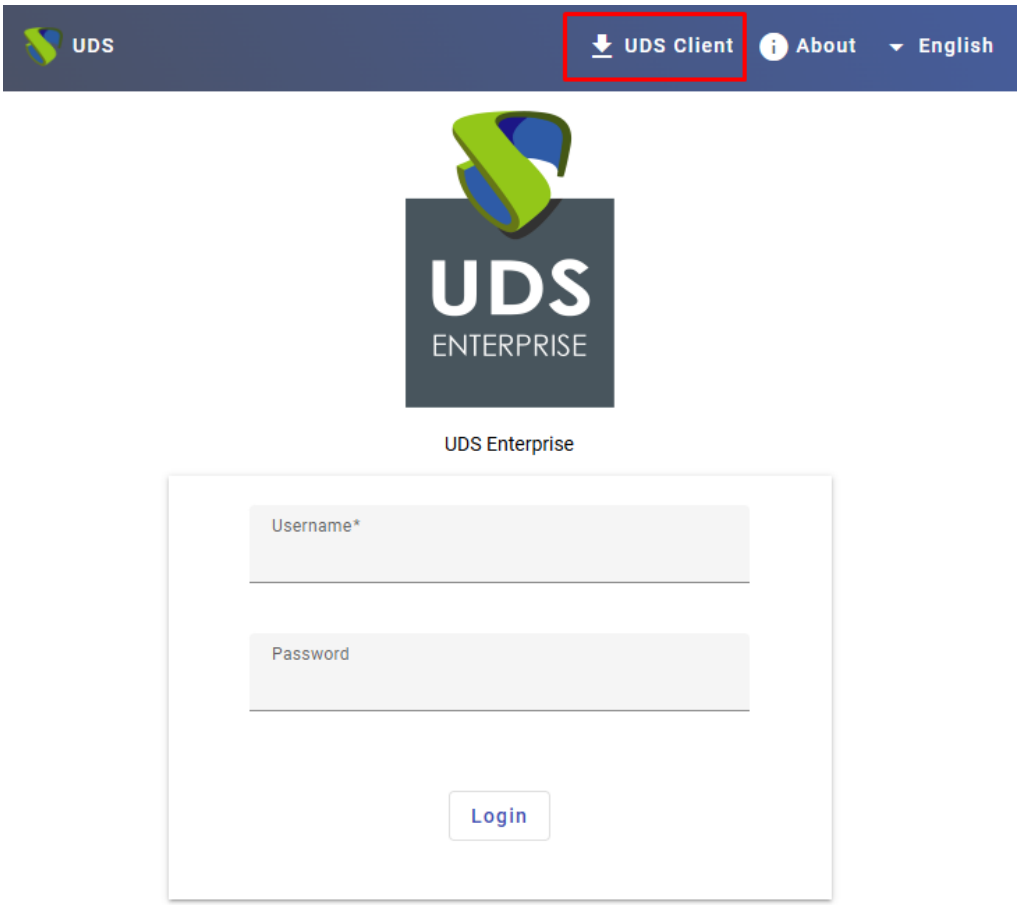
Para proceder a la integración de UDS con Remote Desktop Services puede consultar la guía “Virtualización de aplicaciones Windows con UDS Enterprise” disponible en la sección de documentación de la web de UDS Enterprise.

3.2.5 UDS Client

The UDS Client is a software component that is installed on the connecting client computers that will launch UDS services (virtual desktops, application sessions, etc...). For all connection types supported by UDS, this component will be needed except for the HTML5 connection (which only requires a web browser).

To install the UDS Client, it is necessary to make a previous download from the UDS server (broker) itself, selecting the appropriate client for each type of OS from which we need to access services offered by UDS.

To download, we will access the UDS login portal and in the top menu select "***UDS Client***":

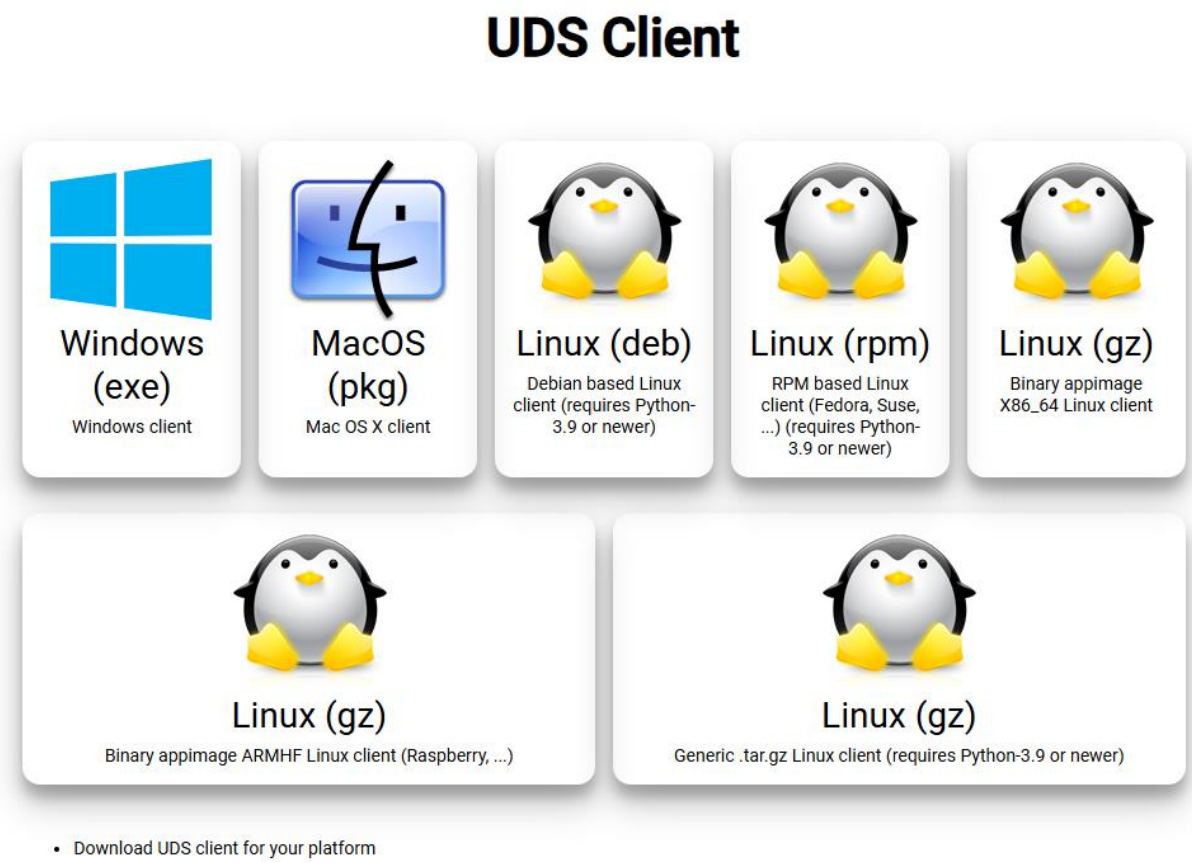


© Virtual Cable S.L.U.

Clients that are available for download will be displayed. We will select the client corresponding to the operating system from which we need to access the different services offered by UDS:

- **Windows client:** UDS client for connection to Windows OS.
- **Mac OS X client:** UDS client for connection with MacOS OS.
- **Debian based Linux client:** UDS client for connection with Debian-based S.O. Linux, such as: Ubuntu, xUbuntu, etc.…
- **RPM based Linux client:** UDS client for connection with Linux OS based on Red Hat, Suse, etc... such as: CentOS, Fedora, etc.…
- **Binary appimage X86_64 Linux client:** Portable UDS client for connection with Linux OS. In addition to the UDS client, it includes the FreeRDP client version 2.3 and the X2Go client

  NOTE: To run the appimage client it will be necessary to have the libfuse2 library installed (libfuse2 for Ubuntu 22, and the corresponding one for the rest of the distributions).
- **Binary appimage ARMHF Linux client:** Portable UDS client for connection with ARM architecture devices. In addition to the UDS client, it includes the FreeRDP client version 2.3 and the X2Go client
- **Genenic .tar.gz Linux client:** UDS client source files for Linux OS

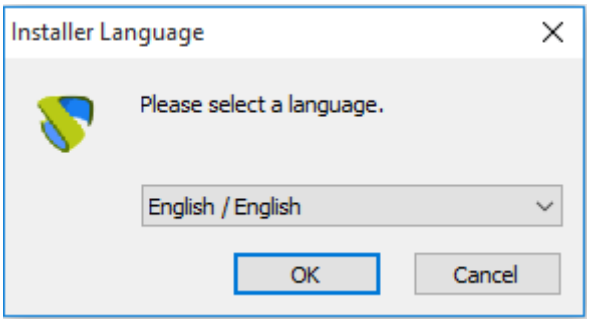


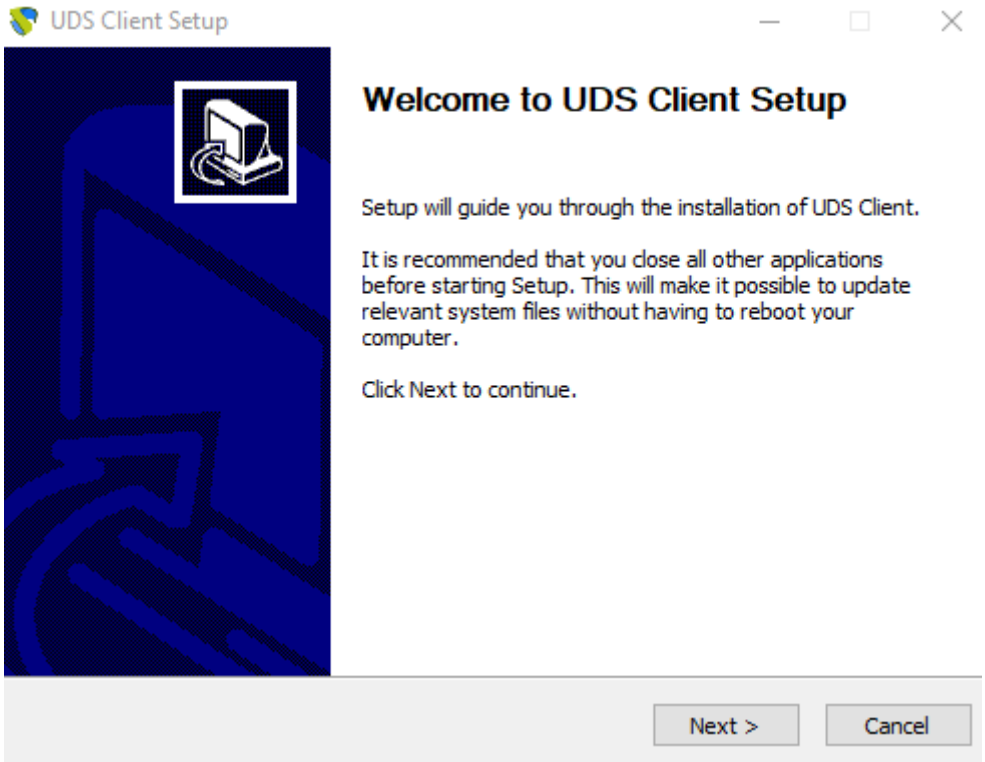
3.2.5.1 Windows

To connect to UDS services from Windows OS connection clients, they must have the UDS client installed: ***UDSClientSetup-4.0.0.exe***

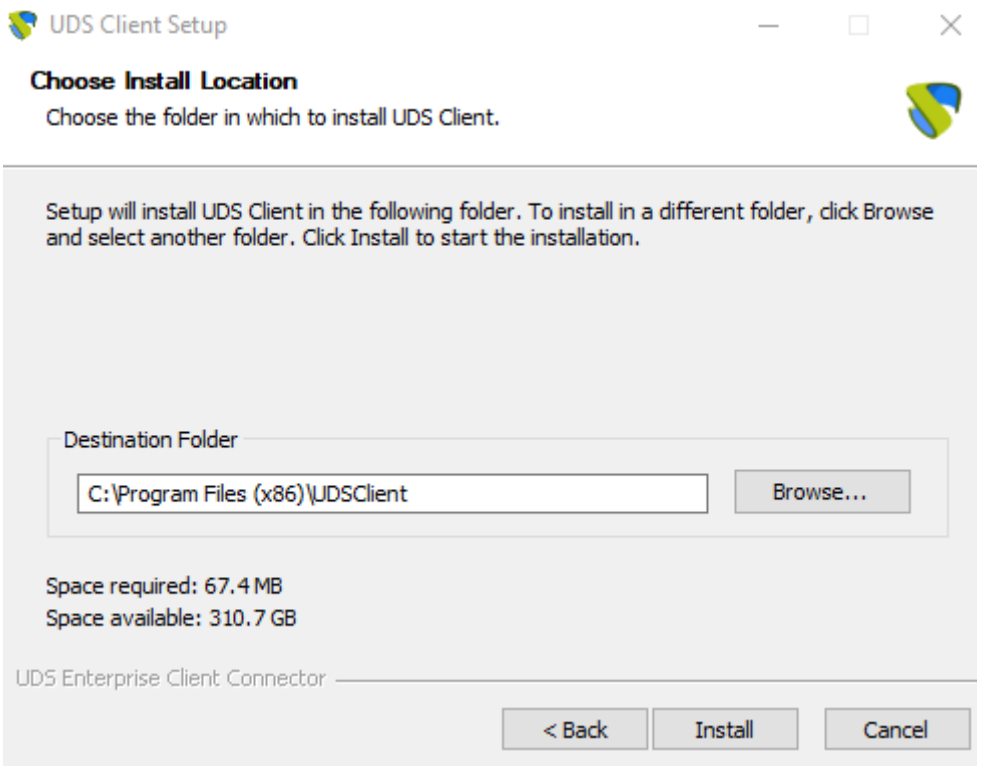
Once the UDS Client for Windows OS has been downloaded, we will run it to proceed with its installation.

Select the installer language:

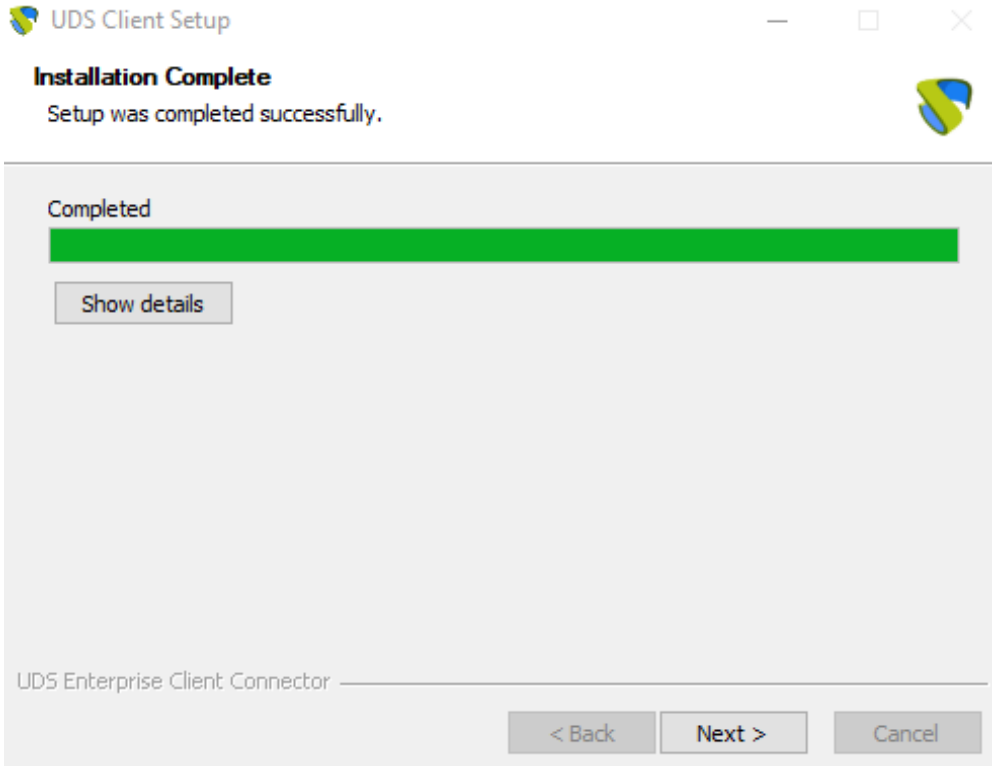




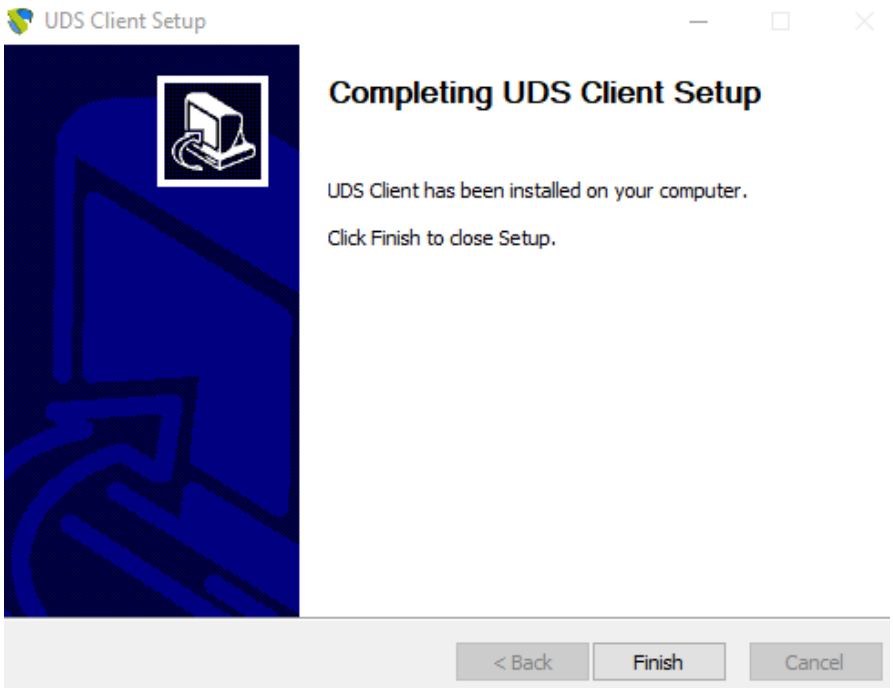
Indicate the installation path of the UDS Client:



Click on "***Install***" to proceed with the installation:



Once the installation is done, we finish the installation wizard:



Once the installation is done, the UDS Client will execute the selected connection protocol (Transport) and connect to the requested service.

- **Unattended installation of the UDS client**

It is possible to perform the unattended installation of the UDS client, for this we will use the parameter: /S

```
C:\Users\user\Downloads>UDSClientSetup-4.0.0.exe /S
```

With this parameter a fully automatic installation will be carried out and even if you already have a UDS client of previous version, it will automatically be uninstalled and the most modern version will be installed.

NOTE:

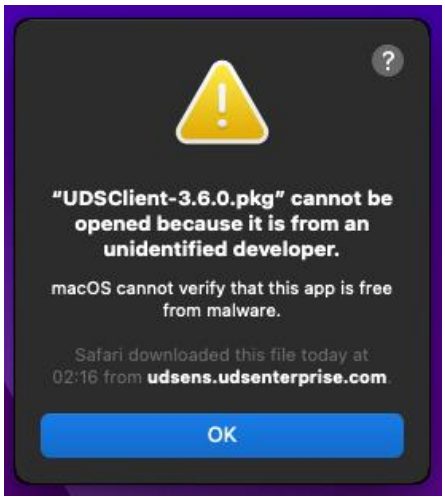
If we access the service with the HTML5 connection type, it will not be necessary to have the UDS Client installed on the connection client computer. The only requirement of this connection is to have a current web browser.

3.2.5.2 MacOS

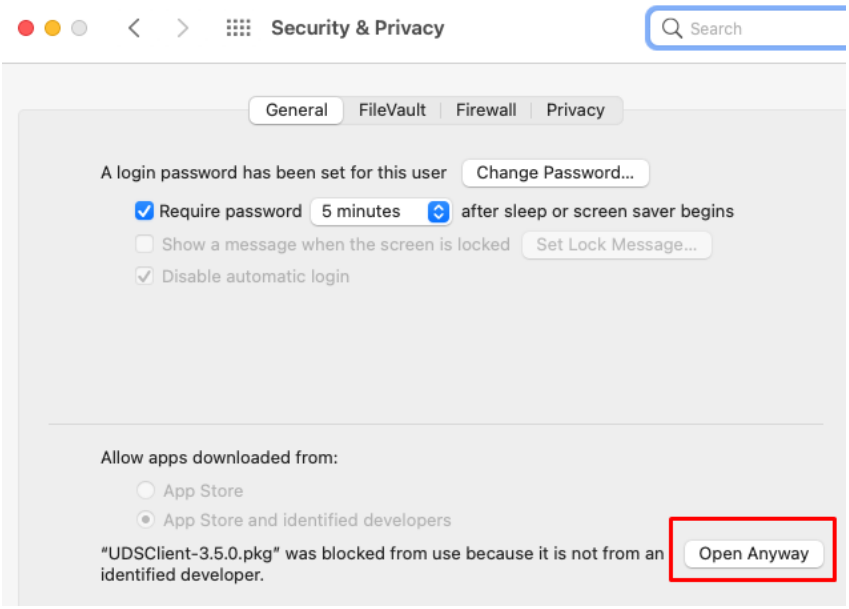
To connect to UDS services from MacOS OS connection clients, they must have the UDS client installed: ***UDSClient-4.0.0.pkg***

Once the UDS Client for S.O. MacOS has been downloaded, we will run it to proceed with its installation.

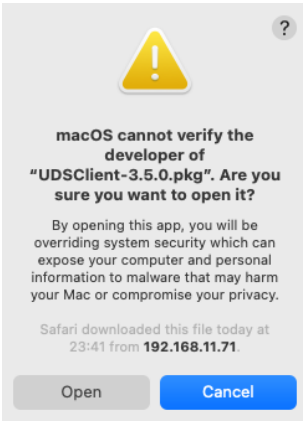
Initially, a warning will appear that prevents us from executing it:



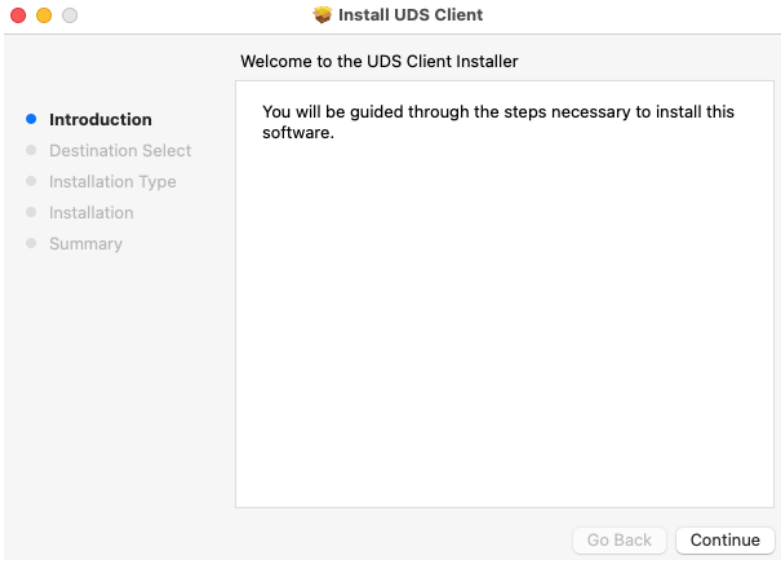
To fix this, we will go to: Apple menu > System Preferences, click Security & Privacy, General:

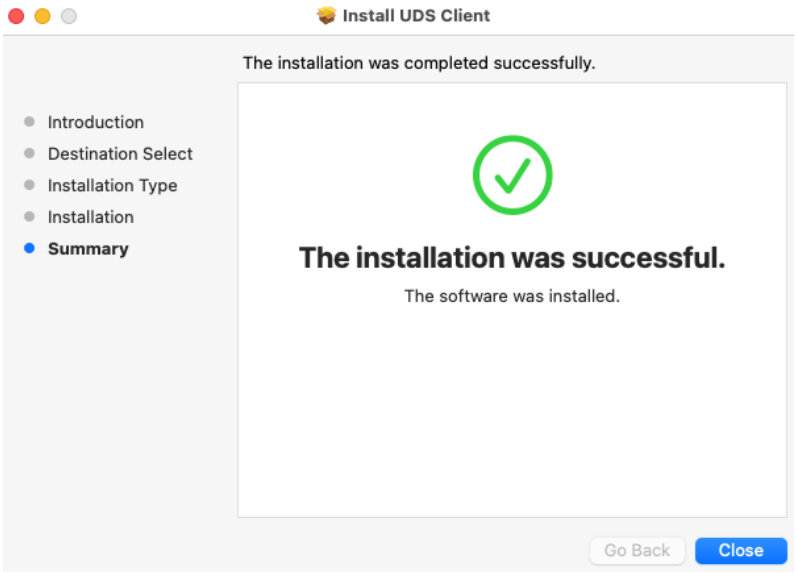


Click on "Open Anyway" to open the UDS Client.



Proceed to the installation of the UDS Client:





Once the installation is done, the UDS Client will execute the RDP protocol client (it must be previously installed) to make the connection with the requested service.

NOTE:

If we access the service with the HTML5 connection type, it will not be necessary to have the UDS Client installed on the connection client computer. The only requirement of this connection is to have a current web browser.

The UDS client can use the RDP clients: FreeRDP or Microsoft Remote Desktop, for more information on its installation and configuration, you can consult the guide: "Enable RDP connection from MacOS devices" available in the documentation section of the UDS Enterprise website.

3.2.5.3 Linux

To connect to UDS services from connection clients with Linux OS, it is necessary that they have the UDS client installed for the different Linux distributions:

- **Debian-based distributions:** ***udsclient3_4.0.0_all.deb***
- **Red Hat and Suse-based distributions:** ***udsclient3-4.0.0-1.noarch.rpm***

Once the UDS Client for the chosen Linux distribution has been downloaded, we will run it to proceed with its installation with administrator permissions.

It is strongly recommended to perform such execution of the UDS Client via command console:

```
user@ubuntu24:~/Descargas$ ls
udsclient3_4.0.0_all.deb
user@ubuntu24:~/Descargas$ sudo dpkg -i udsclient3_4.0.0_all.deb
[sudo] contraseña para user:
Seleccionando el paquete udsclient3 previamente no seleccionado.
(Leyendo la base de datos ... 148904 ficheros o directorios instala
te.)
Preparando para desempaquetar udsclient3_4.0.0_all.deb ...
Desempaquetando udsclient3 (4.0.0) ...
```

If we get an error due to lack of dependencies, we will proceed with its installation:

```
 problemas de dependencias - se deja sin configurar
Procesando disparadores para gnome-menus (3.36.0-1.1ubuntu3) ...
Procesando disparadores para desktop-file-utils (0.27-2build1) ...
Se encontraron errores al procesar:
 udsclient3
user@ubuntu24:~/Descargas$
```

If we get an error due to lack of dependencies, we will proceed with its installation:

```
user@ubuntu24:~/Descargas$ sudo apt install -f
```

```
0 upgraded, 23 newly installed, 0 to remove and 3 not upgraded.
1 not fully installed or removed.
Need to get 16,5 MB of archives.
After this operation, 71,7 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

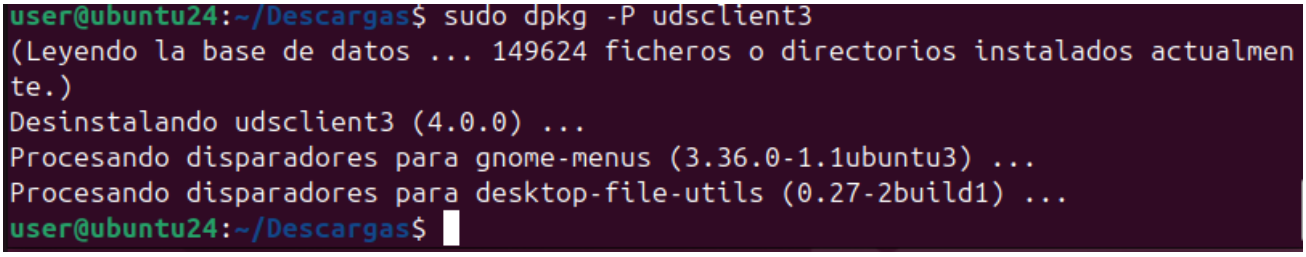
Once the necessary dependencies are installed, the installation is complete. The UDS client will execute the selected connection protocol (Transport) and connect to the requested service.

NOTE:

If we access the service with the HTML5 connection type, it will not be necessary to have the UDS Client installed on the connection client computer. The only requirement of this connection is to have a current web browser.

If we need to uninstall the UDS Client, we will do it with administrator permissions through the command:

```
dpkg -P udsclient3
```



3.2.5.4 Linux AppImage

To make the connection to UDS services from connection clients with Linux OS it is necessary that they have installed the UDS client, this client in its "***Self-content***" mode generated with AppImage will allow us to have in addition to the UDS client: FreeRDP (version 2.3), the X2Go client and Python without having these elements as a requirement to perform its installation.

NOTE:
This UDS AppImage client can be used in almost any Linux distribution (based on Debian, Red Hat, Suse, etc...).

NOTE: To run the appimage client **it will be necessary to have the libfuse2 library installed** (libfuse2 for Ubuntu 22 and the corresponding one for the rest of the distributions).

Once the UDS AppImage: udsclient3-x86_64-4.0.0.tar.gz Client has been downloaded, we unzip it:

```
user@ubuntu24:~/Descargas$ ls
udsclient3-x86_64-4.0.0.tar.gz
user@ubuntu24:~/Descargas$ tar -vxzf udsclient3-x86_64-4.0.0.tar.gz
./
./UDSClient.desktop
./UDSClient-4.0.0-x86_64.AppImage
./installer.sh
user@ubuntu24:~/Descargas$
```

Run the script with administrator permissions ***installer.sh***

```
user@ubuntu24:~/Descargas$ ls
installer.sh                    UDSClient-4.0.0-x86_64.AppImage
udsclient3-x86_64-4.0.0.tar.gz  UDSClient.desktop
user@ubuntu24:~/Descargas$ sudo ./installer.sh
[sudo] contraseña para user:
Installing UDSClient Portable...


******************
*** IMPORTANT! ***
******************
Please, install libfuse2 package in your system to be able to run the AppImage.


Installation process done.
user@ubuntu24:~/Descargas$
```

Once the script installation is finished, we will be able to connect to Windows desktops and virtual applications via RDP protocol (through the FreeRDP client) and to Linux virtual applications via X2Go client

NOTE:

If we access the service with the HTML5 connection type, it will not be necessary to have the UDS Client installed on the connection client computer. The only requirement of this connection is to have a current web browser.

Once the script execution is finished, we will be able to delete the files resulting from unzipping the UDS AppImage client.

4. ABOUT VIRTUAL CABLE

Virtual Cable is a company specialized in the digital **transformation of the workplace**. The company develops, supports and markets UDS Enterprise. It has recently been recognized as an **IDC Innovator in Virtual Client Computing** worldwide. Its team of experts has designed **smart digital workplace solutions** (**VDI, vApp and remote access to physical computers**) tailored to each sector to provide a unique user experience fully adapted to the needs of each user profile. Virtual Cable professionals have **more than 30 years** of experience in IT and software development and more than 15 years in virtualization technologies. **Everyday millions of Windows and Linux virtual desktops** are deployed with UDS Enterprise around the world.

UDS Enterprise is a new software concept for creating a **fully customized workplace virtualization** platform. It provides **secure 24x7 access** from **any location and device** to all applications and software of an organization or educational center.

It allows you to combine Windows and Linux **desktop and application virtualization** in a single console, as well **as remote access** to Windows, Linux and macOS computers. Its Open Source base guarantees **compatibility with any third-party technology**. It can be deployed on-premises, in a public, private, hybrid or **multicloud**. You can even combine several environments at the same time and perform automatic and **intelligent overflows** to optimize performance and efficiency. All with a **single subscription**.

All proper names of programs, operating systems, hardware equipment, etc. appearing herein are trademarks of their respective companies or organizations.

All rights reserved. The content of this work is protected by law, which establishes prison sentences and/or fines, in addition to the corresponding compensation for damages, for those who reproduce, plagiarize, distribute or publicly communicate, in whole or in part, a literary, artistic or scientific work, or its transformation, interpretation or artistic execution fixed in any type of support or communicated through any means, without the required authorization.

-END OF DOCUMENT-